

Computational Fluid Dynamics

FLUID DYNAMICS CONFERENCE

COMPUTATIONAL

(NASA-CP-10078) COMPUTATIONAL FLUID
DYNAMICS (NASA) 219 p

N92-25808

--THRU--

N92-25827

Unclas

G3/07 0091360

*Lewis presentations at the
NASA CFD Conference hosted by
NASA Ames Research Center
Moffett Field, California
March 12 to 14, 1991*

NASA

NASA Conference Publication 10078

Computational Fluid Dynamics

*Lewis presentations at the
NASA CFD Conference hosted by
NASA Ames Research Center
Moffett Field, California
March 12 to 14, 1991*



National Aeronautics and
Space Administration

Office of Management

**Scientific and Technical
Information Program**

1992

PREFACE

The first NASA Computational Fluid Dynamics (CFD) Conference was held at NASA Ames Research Center on March 7 through 9, 1989. Conference objectives were to disseminate CFD results to industry and university researchers, to promote synergy among NASA CFD researchers, and to permit feedback from researchers outside of NASA concerning issues facing the discipline of CFD. The focus of the 1989 conference was on the application of CFD technology. As a result of three panel sessions held at the 1989 conference, the following conclusions were reached:

- NASA's program was too heavily focused on applications.
- CFD developers need to better understand the needs of the users.
- Industry needs more reliable and cost effective CFD tools.
- Three critical areas of research were identified:
 - More algorithm research, particularly for Navier-Stokes solvers with unstructured grids.
 - More research on geometric modeling with rapid, accurate, and effective surface definition methods.
 - More research on grid generation stressing faster, more efficient, and improved quality with emphasis on reduced set-up time and complexity.

Additional concerns resulting from the panels may be found in NASA Conference Publication 10038, Vol. 1, p. xiv.

The objectives of the second NASA CFD Conference, held at the Ames Research Center on March 12 through 14, 1991, were the same as the 1989 conference: to disseminate CFD research results to industry and university CFD researchers, to promote synergy among NASA CFD researchers, and to permit feedback from researchers outside NASA on issues facing the discipline of CFD. However, the focus of the 1991 conference was on the elements that comprise the discipline of CFD, rather than on CFD applications. These elements are

- Algorithm development and analysis
- Algorithms for real gas/combustion modeling
- Algorithms for advanced computer architectures
- Turbulence modeling/flow physics
- Scientific visualization
- Grid generation
- Other CFD methodologies

The intent of the 1991 conference was to emphasize CFD development underway at NASA Centers, rather than the applications activity.

This publication is a collection of 20 presentations made by members of the NASA Lewis Research Center staff; the members of the Institute for Computational Mechanics in Propulsion (ICOMP); Sverdrup Corporation, Lewis Research Group; and visiting academics at Lewis. Presentation abstracts, (published in bound form at the conference) as well as viewgraph material used during the conference are included in this publication. The intent is to display research underway on the fundamentals of CFD at Lewis.

The conference was sponsored by the Aerodynamics Division, Office of Aeronautics, Exploration and Technology (OAET), NASA Headquarters, Washington, D.C.

This preface is based on an adaptation of that appearing in NASA Conference Publication 10038, proceedings of the CFD Conference held at Ames Research Center, 1989.

Louis A. Povinelli
NASA Lewis Research Center
April 8, 1991

CONTENTS

Computational Fluid Dynamics at the Lewis Research Center- An Overview

L.A. Povinelli	1
--------------------------	---

ALGORITHM DEVELOPMENT -- UPWIND/TVD/ENO METHODS

Development of New Flux Splitting Schemes

M.-S. Liou and C.J. Steffan, Jr.	19
--	----

ALGORITHM DEVELOPMENT -- NEW METHODS AND ANALYSIS

An Iterative Implicit DDADI Algorithm for Solving the Navier-Stokes Equation

S.C. Chen, N.S. Liu, and H.D. Kim	29
---	----

ALGORITHM DEVELOPMENT -- CODES AND METHODS

The Proteus Navier-Stokes Code

C.E. Towne and J.R. Schwab	47
--------------------------------------	----

Accurate Upwind-Monotone (Nonoscillatory) Methods for Conservation Laws

H.T. Huynh	57
----------------------	----

Numerical Simulation of Conservation Laws

S. Chung-Chang and W.M. To	65
--------------------------------------	----

ALGORITHMS FOR REAL GAS

A New Lagrangian Method for Real Gases at Supersonic Speed

C.Y. Loh and M.-S. Liou	75
-----------------------------------	----

A Time-Accurate Implicit Method for Chemical Non-Equilibrium Flows at All Speeds

J.-S. Shuen	83
-----------------------	----

Study of Shock-Induced Combustion Using an Implicit TVD Scheme

S. Yungster	93
-----------------------	----

Upwind Schemes and Bifurcating Solutions in Real Gas Computations

A. Suresh and M.-S. Liou	103
------------------------------------	-----

ALGORITHMS FOR ADVANCED ARCHITECTURES

High Order Parallel Numerical Schemes for Solving Incompressible Flows

A. Lin, E.J. Milner, M.-F. Liou, and R.A. Belch	115
---	-----

An Efficient and Robust Algorithm for Time Dependent Viscous Incompressible Navier-Stokes Equations	
J.W. Goodrich	133

TURBULENCE MODELING/FLOW PHYSICS

Advances in Engineering Turbulence Modeling	
T.-H. Shih	143
On Recontamination and Directional-Bias Problems in Monte Carlo Simulation of PDF Turbulence Models	
A.T. Hsu	163
Implementation of a $k-\epsilon$ Turbulence Model to RPLUS3D Code	
T. Chitsomboon	175
Turbulence and Deterministic Chaos	
R.G. Deissler	181
Stability of Compressible Taylor-Couette Flow	
K. Kao and C. Chow	193

SCIENTIFIC VISUALIZATION

Techniques for Animation of CFD Results	
J.G. Horowitz and J.C. Hanson	205
Distributed Visualization for Computational Fluid Dynamics	
D.J. Sosoka and A.A. Facca	219

GRID GENERATION

Techniques for Grid Manipulation and Adaptation	
Y.K. Choo, P.R. Eisemann, and K.D. Lee	225



LEWIS RESEARCH CENTER

**COMPUTATIONAL FLUID DYNAMICS
AT THE
LEWIS RESEARCH CENTER**

- AN OVERVIEW

**LOUIS A. POVINELLI
DEPUTY CHIEF
INTERNAL FLUID MECHANICS DIVISION**

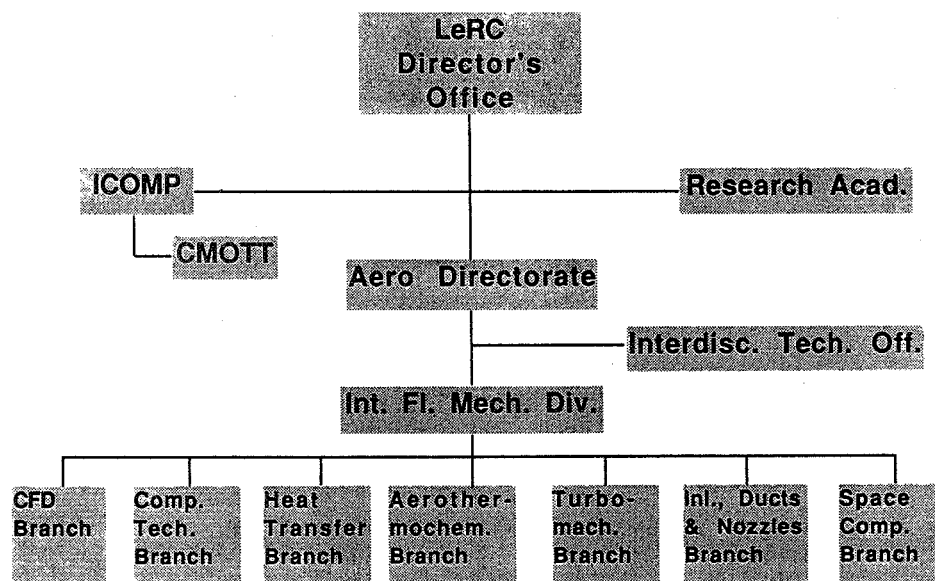
INTERNAL FLUID MECHANICS DIVISION

OVERVIEW OUTLINE

- ORGANIZATIONAL STRUCTURE
- MAIN THRUSTS
- LEVEL OF EFFORT
- COMPUTATIONAL FACILITIES
- TECHNICAL HIGHLIGHTS
- FUTURE DIRECTIONS

INTERNAL FLUID MECHANICS DIVISION

ORGANIZATIONAL STRUCTURE



INTERNAL FLUID MECHANICS DIVISION

INSTITUTE FOR COMPUTATIONAL MECHANICS IN PROPULSION (ICOMP)

OBJECTIVE: TO DEVELOP ADVANCED COMPUTATIONAL METHODS REQUIRED FOR THE SOLUTION OF INTERNAL FLUID MECHANICS AND STRUCTURAL MECHANICS PROBLEMS

ORGANIZATION

PROGRAM DIRECTOR: Dr. Louis Povinelli

EXECUTIVE OFFICER: Dr. Charles Feller

STEERING COMMITTEE: Prof. E. Reshotko, Chairman

Dr. H. Brandhorst
Dr. M. Goldstein
Prof. R. Mullen
Dr. L. Plinson

Dr. L. Diehl
Prof. I. Greber
Dr. L. Nichols
Dr. L. Reid

INTERNAL FLUID MECHANICS DIVISION



INSTITUTE FOR COMPUTATIONAL MECHANICS IN PROPULSION

ICOMP STATISTICS

	<u>1986</u>	<u>1987</u>	<u>1988</u>	<u>1989</u>	<u>1990</u>
PEOPLE	23	43	50	46	47
SEMINARS	10	27	39	30	37
REPORTS	2	9	22	32	25
WORKSHOPS/LECTURE SERIES	1	0	2	1	1
PRESENTATIONS	7	0	21	14	15
UNIVERSITIES OR ORGANIZATIONS	20	28	43	35	36

Figure 4. - ICOMP statistics (1986 to 1990)



INSTITUTE FOR COMPUTATIONAL MECHANICS IN PROPULSION

ICOMP/LeRC JOINT ACTIVITIES

<u>ICOMP MEMBER</u>	<u>LeRC CIVIL SERVANT</u>	<u>ACTIVITY</u>
AFOLABI (PURDUE-IND)	STEFKO	BLADE AEROELASTICITY-MODAL INTERACTIONS
AGGARWAL (ILL-CHICAGO)	BULZAN	SPRAY COMBUSTION MODELS
AKL (LOUISIANA TECH)	KIRALY	FINITE ELEMENT MODEL MAPPING ONTO PARALLEL MICROCOMPUTING ARCHITECTURE
ARNONE (FLORENCE)	LIQU	TRANSONIC FLOW CALCULATIONS/NON- PERIODIC C-GRIDS
BALSA (ARIZONA)	GOLDSTEIN	INSTABILITIES IN SUPERSONIC MIXING LAYERS
COWLEY (CAMBRIDGE)	HULTGREN	NON-LINEAR BOUNDARY LAYER PROBLEMS- SEPARATION
DEMUREN (OLD DOM)	CLAUS	FULL REYNOLDS STRESS TURBULENCE MODELS
EISEMAN (COLUMBIA)	CHOD	ADAPTIVE GRID GENERATION
GAJJAR (EXETER)	GOLDSTEIN	UNSTEADY NON-LINEAR CRITICAL LAYERS
HAGSTROM (NEW MEXICO)	GOODRICH	OUTFLOW BOUNDARY CONDITIONS IN LOW MACH NO. COMBUSTION



INSTITUTE FOR COMPUTATIONAL MECHANICS IN PROPULSION

ICOMP STATISTICS

	<u>1986</u>	<u>1987</u>	<u>1988</u>	<u>1989</u>	<u>1990</u>
PEOPLE	23	43	50	46	47
SEMINARS	10	27	39	30	37
REPORTS	2	9	22	32	25
WORKSHOPS/LECTURE SERIES	1	0	2	1	1
PRESENTATIONS	7	0	21	14	15
UNIVERSITIES OR ORGANIZATIONS	20	28	43	35	36

Figure 4. - ICOMP statistics (1986 to 1990)

CENTER FOR MODELING OF TURBULENCE AND TRANSITION (CMOTT)

TECHNICAL LEADER: DR. T. SHIH

OBJECTIVE: DEVELOP VALIDATED ENGINEERING CLOSURE MODELS FOR IMPLEMENTATION IN PROPULSION CFD

SCOPE: MODELING OF COMPRESSIBLE TURBULENT FLOWS

- TWO-EQUATION MODELS
- SECOND ORDER CLOSURE MODEL
- RENORMALIZATION GROUP THEORY (RNG)
- DIRECT INTERACTION APPROXIMATION (DIA)

TURBULENCE/CHEMISTRY INTERACTION

- PDF PLUS MOMENT CLOSURE MODELS

BOUNDARY LAYER TRANSITION

- MODELING OF BYPASS TRANSITION
- IMPROVEMENT OF NEAR WALL TURBULENCE MODELS
- DNS OF BYPASS TRANSITION

INTERNAL FLUID MECHANICS DIVISION

INTERNAL FLUID MECHANICS DIVISION

Dr. Lonnie Reid, Chief

2600

Dr. Louis A. Povinelli, Deputy Chief

**COMPUTATIONAL FLUID DYNAMICS
BRANCH** 2610
Dr. D. Raam Reddy, Chief

**HEAT TRANSFER
BRANCH** 2630
Dr. Robert J. Simoneau, Chief
Frederick F. Simon, Deputy Chief

**AEROTHERMOCHEMISTRY
BRANCH** 2650
Dr. Edward J. Mularz, Chief
Erwin A. Lezberg, Deputy Chief

**COMPUTATIONAL METHODS FOR SPACE
BRANCH** 2670
Dr. Robert M. Stubbs, Chief

**COMPUTATIONAL TECHNOLOGY
BRANCH** 2620
John R. Szuch, Chief
Richard A. Blech, Deputy Chief

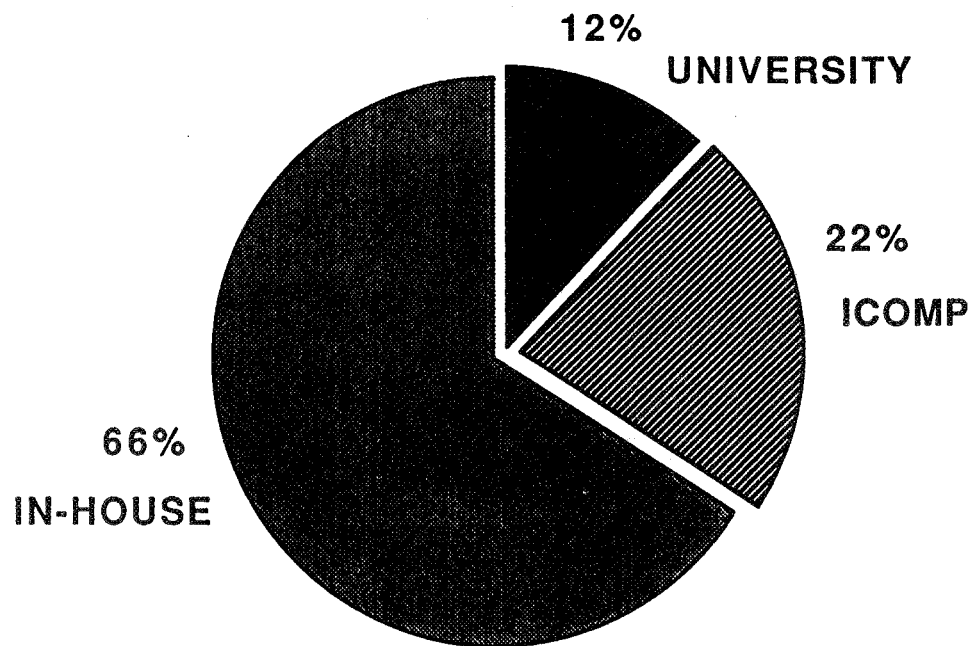
**TURBOMACHINERY FLOW PHYSICS
BRANCH** 2640
Dr. Raymond E. Gaugler, Chief
Jerry R. Wood, Deputy Chief

**INLET, DUCT & NOZZLE FLOW PHYSICS
BRANCH** 2660
John M. Abbott, Chief
Dr. James R. Scott, Deputy Chief

INTERNAL FLUID MECHANICS DIVISION

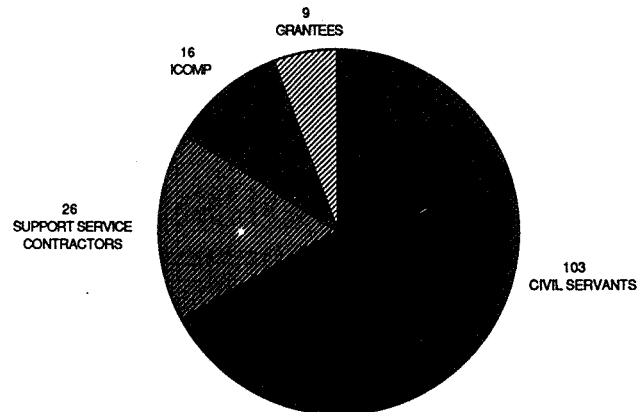
FUNDING DISTRIBUTION

BASE RTOP (505-62-52)



LEVEL OF EFFORT

IFMD PERSONNEL



INTERNAL FLUID MECHANICS DIVISION

LeRC COMPUTATIONAL RESOURCES ON NETWORK

- CRAY Y-MP (6 proc., 128 MW)
- CRAY X-MP (2 proc., 8 MW)
- CONVEX C220 (2 proc., 256 MB)
- SCIENTIFIC VAXcluster (5 proc., 448 MB)
- VM/CMS/UTS (2 proc., 128 MB)
- GRAPHICS & VISUALIZATION (G-VIZ) LAB
- ADVANCED COMPUTATIONAL CONCEPTS LAB (ACCL)
 - INTEL IPSC/860 (16 nodes, 256 MB)
 - ALLIANT FX/80 (8 proc., 128 MB)
 - STELLAR GS1000 & IRIS 4D/320 (2 proc.)

INTERNAL FLUID MECHANICS DIVISION

IFMD COMPUTATIONAL RESOURCES

- HIGH-END GRAPHICS WORKSTATIONS (11)
- PERSONAL GRAPHICS AND COMPUTER WORKSTATIONS (30)
- PERSONAL COMPUTERS (26)
- COLOR HARD-COPY UNITS (16)
- ADMINISTRATION OF NAS ACCOUNTS (26,000 HRS.)
 - IN-HOUSE, ~50%
 - UNIVERSITY, INDUSTRY, ~50%

INTERNAL FLUID MECHANICS DIVISION

IFMD GOAL**DEVELOP VALIDATED, NUMERICAL SIMULATIONS FOR PROPULSION SYSTEMS**

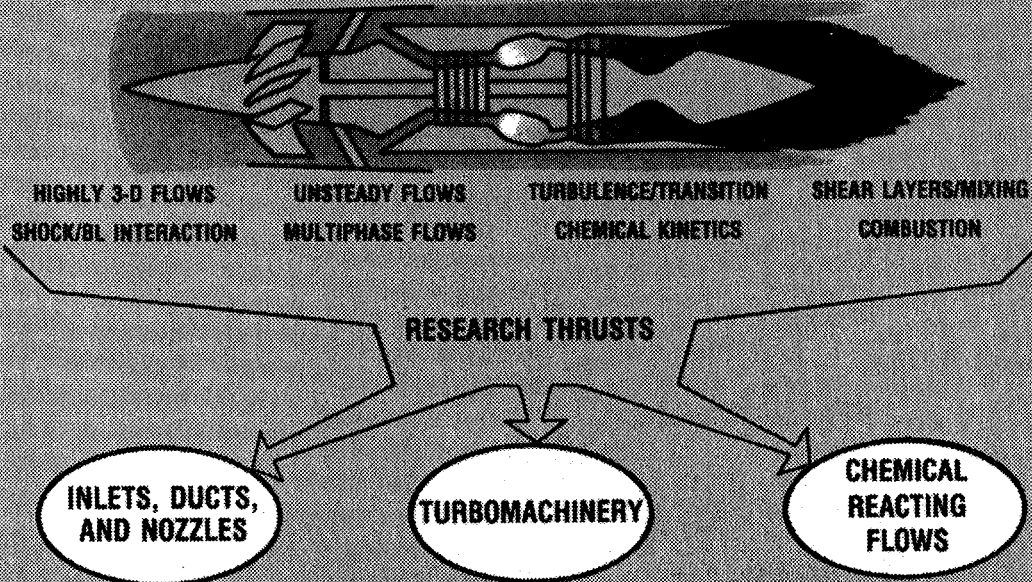
- GAS TURBINES
- ROCKETS
- HYPERSONIC AIR BREATHING SYSTEMS

BY CONDUCTING CLOSELY COUPLED EXPERIMENTAL/COMPUTATIONAL RESEARCH ON:

- INLETS, DUCTS, NOZZLES
- COMBUSTORS
- TURBOMACHINERY COMPONENTS

INTERNAL FLUID MECHANICS DIVISION

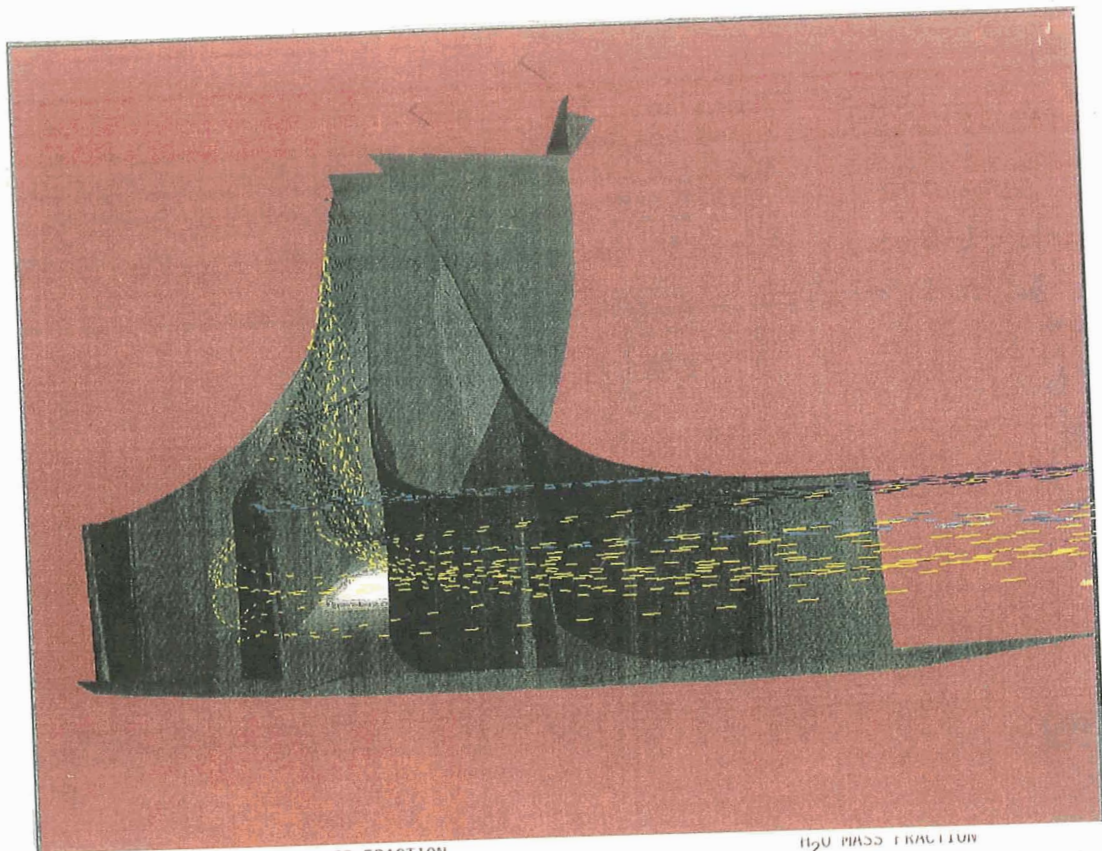
RESEARCH THRUSTS



CD-89-21321

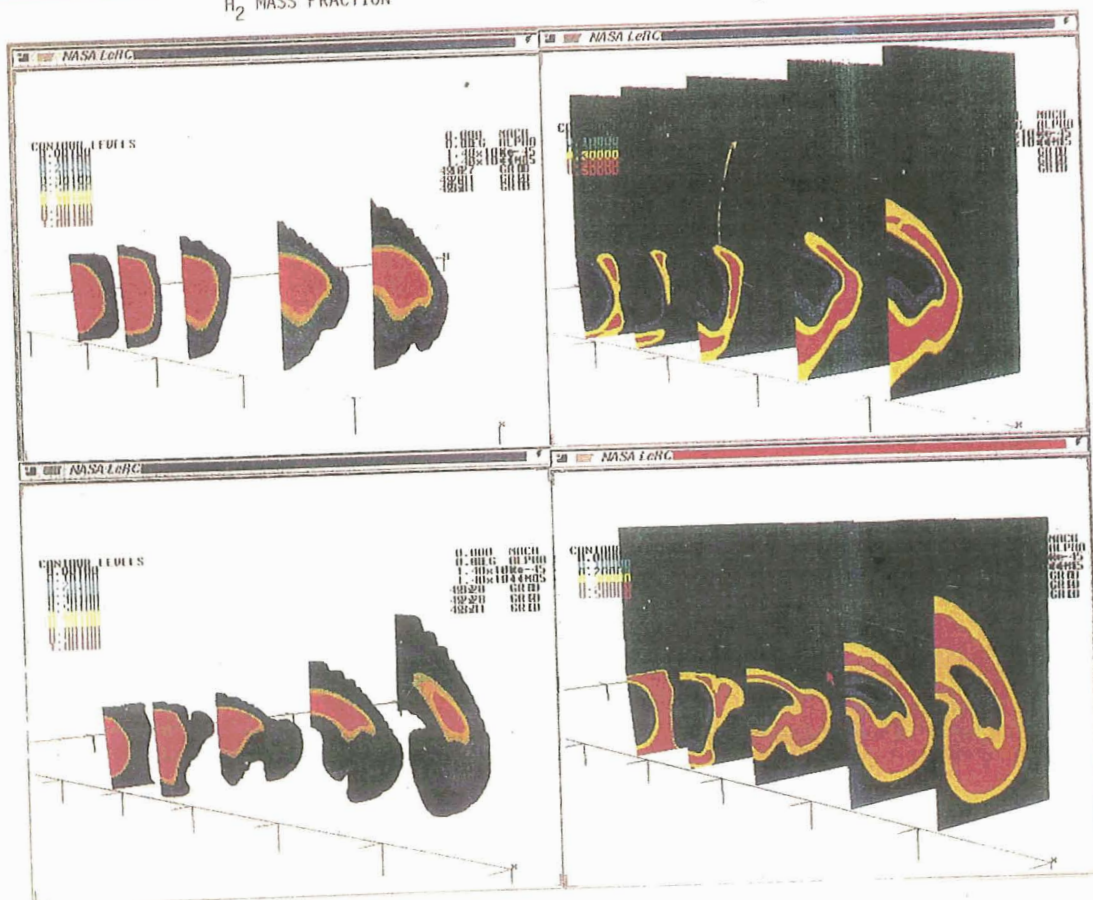
IFMD MAIN THRUSTS

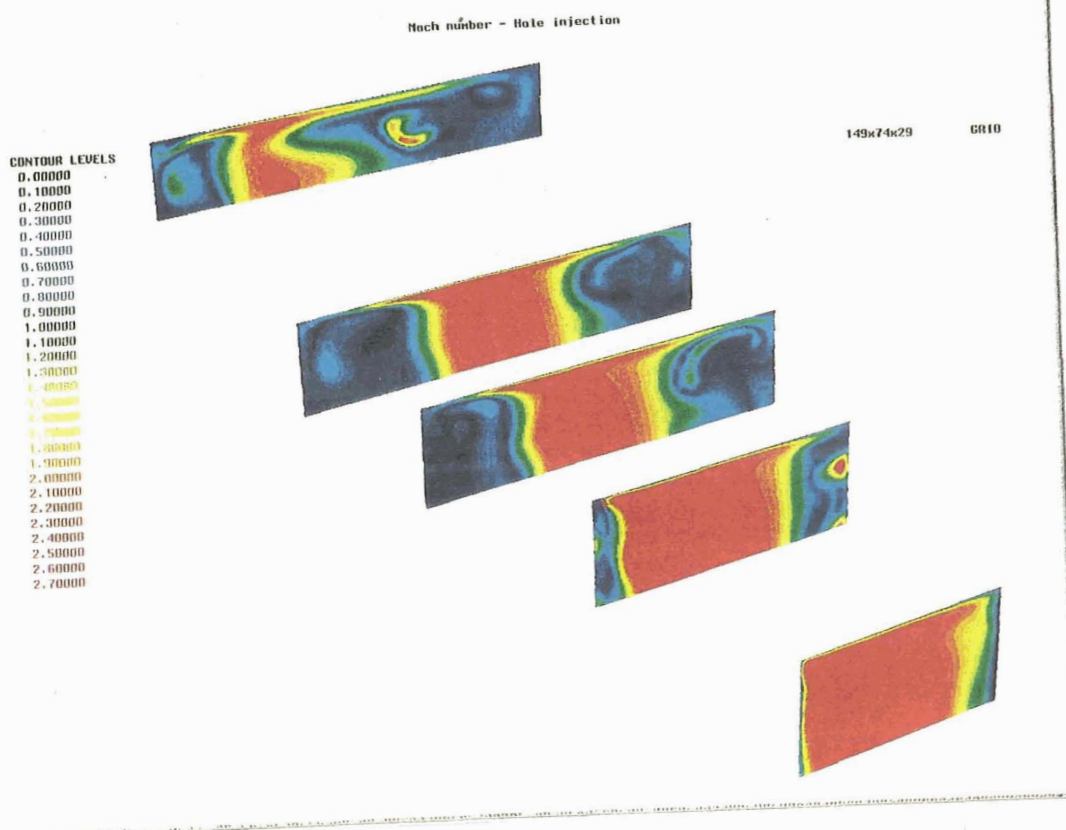
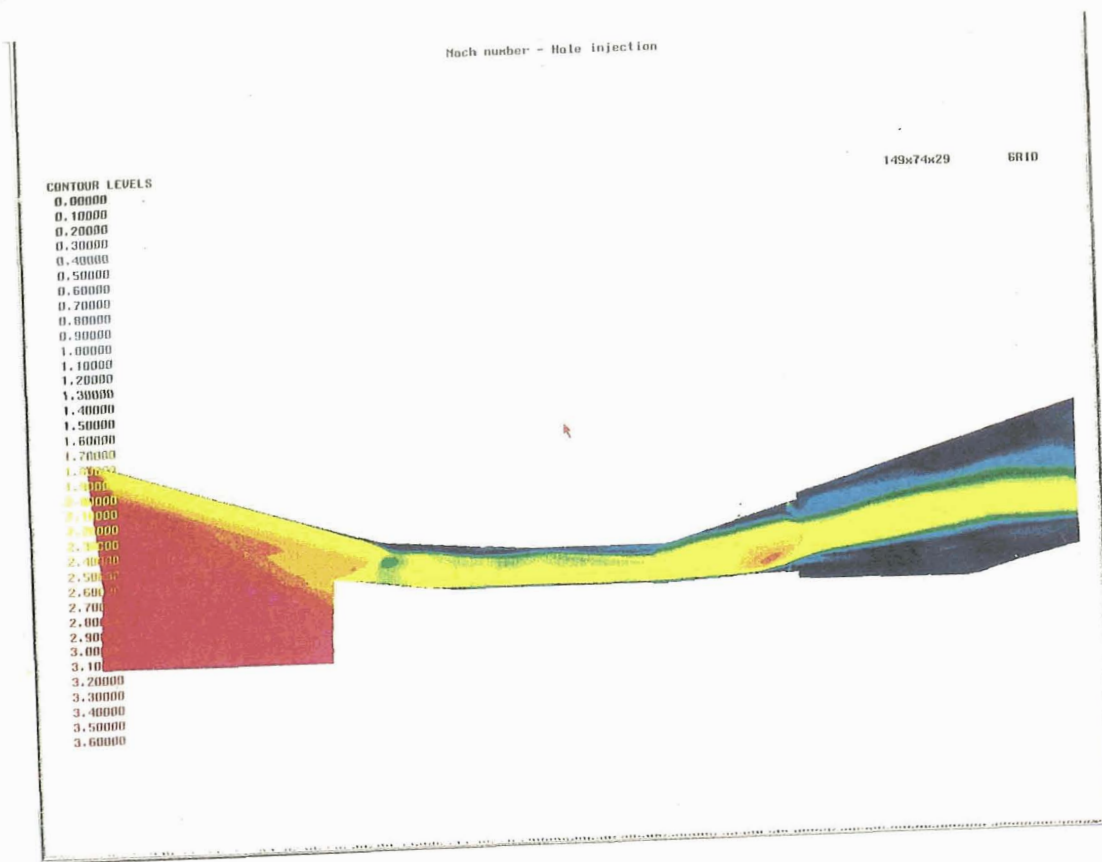
- **ALGORITHM DEVELOPMENT**
 - NAVIER-STOKES
 - LOW SPEED COMPRESSIBLE
 - BOLTZMANN & DSMC
- **CLOSURE MODELING**
 - TURBULENCE & TRANSITION
 - PDF MODELING FOR REACTING FLOWS
- **REAL-TIME INTEGRATION OF CFD/EXPERIMENT (ICE)**



H₂ MASS FRACTION

H₂O MASS FRACTION





LeRC CFD CONFERENCE PAPERS

Algorithm Development

- M. Liou, C. Steffen, "Development of New Flux Splitting Schemes"
S. Chen, N. Liu, H. Kim, "An Iterative Implicit DDADI Algorithm for Solving the Navier-Stokes Equations"
C. Towne, J. Schwab, "The PROTEUS Navier-Stokes Code"
H. Huynh, "Accurate Upwind-Monotone Methods for Conservation Laws"
S. Chang, W. To, "Numerical Simulation of Conservation Laws"

Algorithms For Real Gas

- C. Loh, M. Liou, "A New Lagrangian Method for Real Gases at Supersonic Speed"
J. Shuen, "A Strongly Implicit Method for Chemical Non-Equilibrium Viscous Flows at all Speeds"
S. Yungster, "Study of Shock-Induced Combustion Using An Implicit TVD Scheme"
A. Suresh, M. Liou, "Upwind Schemes and Bifurcating Solutions in Real Gas Computations"

INTERNAL FLUID MECHANICS DIVISION

LeRC CFD CONFERENCE PAPERS

Algorithms for Advanced Architectures

- A. Lin, E. Milner, M. Liou, R. Blech, "High Order Parallel Numerical Scheme for Solving Incompressible Flows"
J. Goodrich, "An Efficient and Robust Algorithm for Time Dependent Viscous Incompressible Navier-Stokes Equations"

Turbulence Modeling/Flow Physics

- T. Shih, "Advances in Engineering Turbulence Modeling"
A. Hsu, "On the Recontamination Problem in Monte Carlo Simulation of PDF Turbulence Models"
T. Chitsomboon, "Implementation of a K- ϵ Turbulence Model to RPLUS3D Code"
R. Deissler, "Turbulence and Deterministic Chaos"
K. Kao, C. Chow, "Stability of Compressible Taylor-Couette Flow"

Scientific Visualization

- J. Horowitz, J. Hanson, "Techniques for Animation of CFD Results"
D. Sosoka, T. Facca, "Distributed Visualization for CFD"

INTERNAL FLUID MECHANICS DIVISION

LeRC CFD CONFERENCE PAPERS**Algorithms for Advanced Architectures**

- A. Lin, E. Milner, M. Liou, R. Blech, "High Order Parallel Numerical Scheme for Solving Incompressible Flows"
J. Goodrich, "An Efficient and Robust Algorithm for Time Dependent Viscous Incompressible Navier-Stokes Equations"

Turbulence Modeling/Flow Physics

- T. Shih, "Advances in Engineering Turbulence Modeling"
A. Hsu, "On the Recontamination Problem in Monte Carlo Simulation of PDF Turbulence Models"
T. Chitsomboon, "Implementation of a K- ϵ Turbulence Model to RPLUS3D Code"
R. Deissler, "Turbulence and Deterministic Chaos"
K. Kao, C. Chow, "Stability of Compressible Taylor-Couette Flow"

Scientific Visualization

- J. Horowitz, J. Hanson, "Techniques for Animation of CFD Results"
D. Sosoka, T. Facca, "Distributed Visualization for CFD"

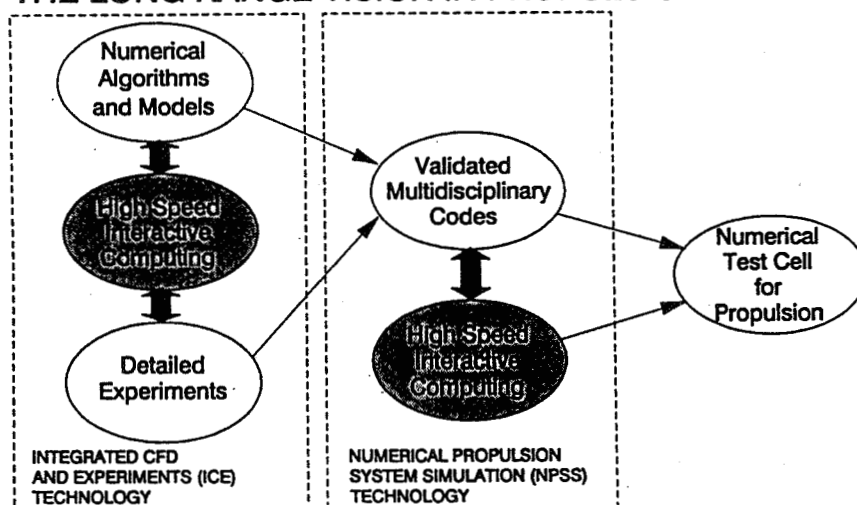
INTERNAL FLUID MECHANICS DIVISION

LeRC CFD CONFERENCE PAPERS**Grid Generation**

- Y. Choo, P. Eiseman, K. Lee, "Techniques for Grid Manipulation and Adaptation"

INTERNAL FLUID MECHANICS DIVISION

THE LONG RANGE VISION IN PROPULSION ANALYSIS



Development of New Flux Splitting Schemes

Meng-Sing Liou and Christopher J. Steffen, Jr.
Internal Fluid Mechanics Division
NASA Lewis Research Center

Maximizing both accuracy and efficiency has been the primary objective in designing a numerical algorithm for computational fluid dynamics (CFD). This is especially important for solution of complex 3D systems of Navier-Stokes equations which often include turbulence modeling and chemistry effects. Recently, upwind schemes have been well received for both their capability of resolving discontinuities and their sound theoretical basis in characteristic theory for hyperbolic systems. With this in mind, we present two new flux splitting techniques for upwind differencing.

The first method is based upon High-Order Polynomials Expansions (HOPE) of the mass flux vector. The present splitting results in positive and negative mass-flux components that vanish at $M = 0$. Thus the error in the Van Leer scheme which results in the diffusion of the boundary layer is eliminated. We also introduce several choices for splitting the pressure and examine their effects on the solution.

The second new flux splitting is based on the Advection Upwind Splitting Method (AUSM for short). In Navier-Stokes calculations, the diffusion error present in Van Leer's flux-splitting scheme corrupts the velocity vector near the wall. In the AUSM, a proper splitting of the advective velocity component leads to an accurate resolution of the interface fluxes. The interface velocity is defined using the Mach number polynomial expansion in the mass flux, then the convective fluxes follow directly. Again, several choices of pressure splitting are possible among which a simple Mach number splitting according to characteristics appears to be the best in terms of accuracy. The scheme has yielded results whose accuracy rivals, and in some cases surpasses that of Roe's method, at reduced complexity and computational effort.

The calculation of the hypersonic conical flow demonstrates the accuracy of the splittings in resolving the flow in the presence of strong gradients. The second series of tests involving the 2D inviscid flow over a NACA 0012 airfoil demonstrate the ability of the AUSM to resolve the shock discontinuity at transonic speed and the level of entropy generation at the stagnation point.

In the third case we calculate a series of supersonic flows over a circular cylinder. The Roe splitting in all conditions and grids tested yielded anomalous solutions (sometimes referred to as the carbuncle phenomenon), which could appear as nonsymmetric, protuberant, or indented contours, see Figs. 1. The AUSM gave expected solutions in all calculations.

The fourth test deals with a 2D shock wave/boundary layer interaction. This provides an opportunity to accurately resolve a laminar separation region and to compare the ability to resolve a non grid-aligned shock with other methods, as shown in Figs. 2.

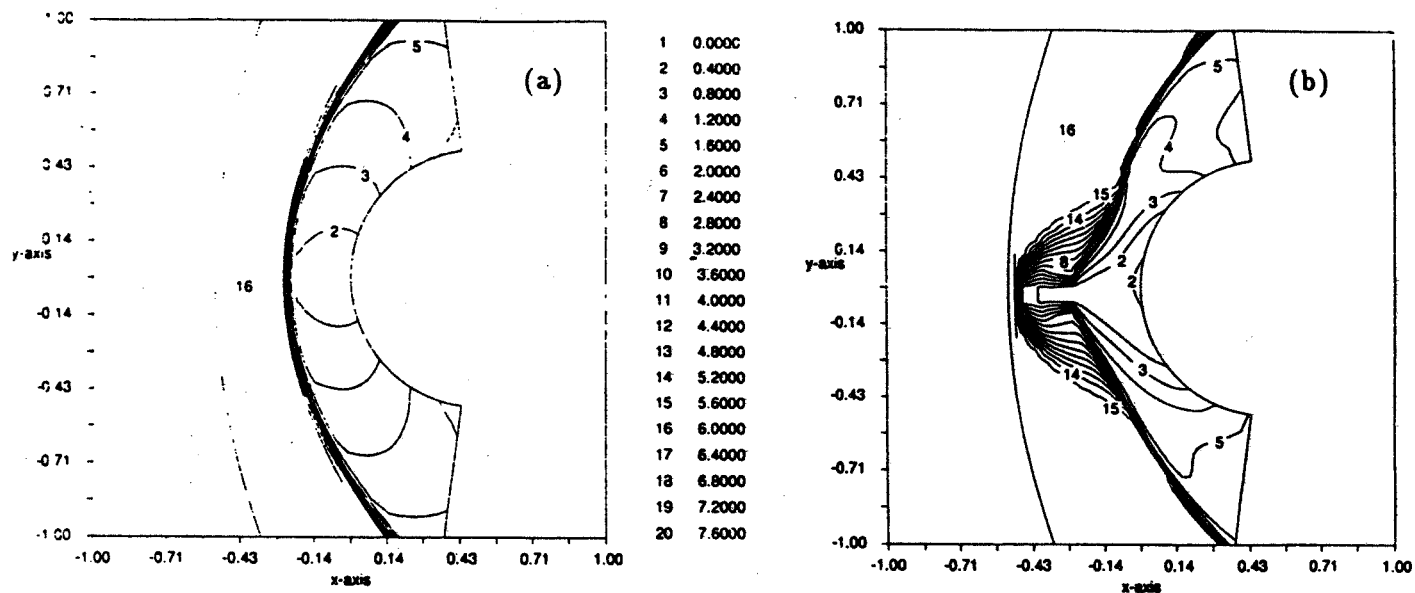


Fig. 1 Mach contours for a Mach 6 flow over a circular cylinder: (a) the AUSM solution, and (b) the Roe solution displaying a protuberant, two-shock solution. Note the calculations were performed for the front half plane, thus no symmetry was imposed.

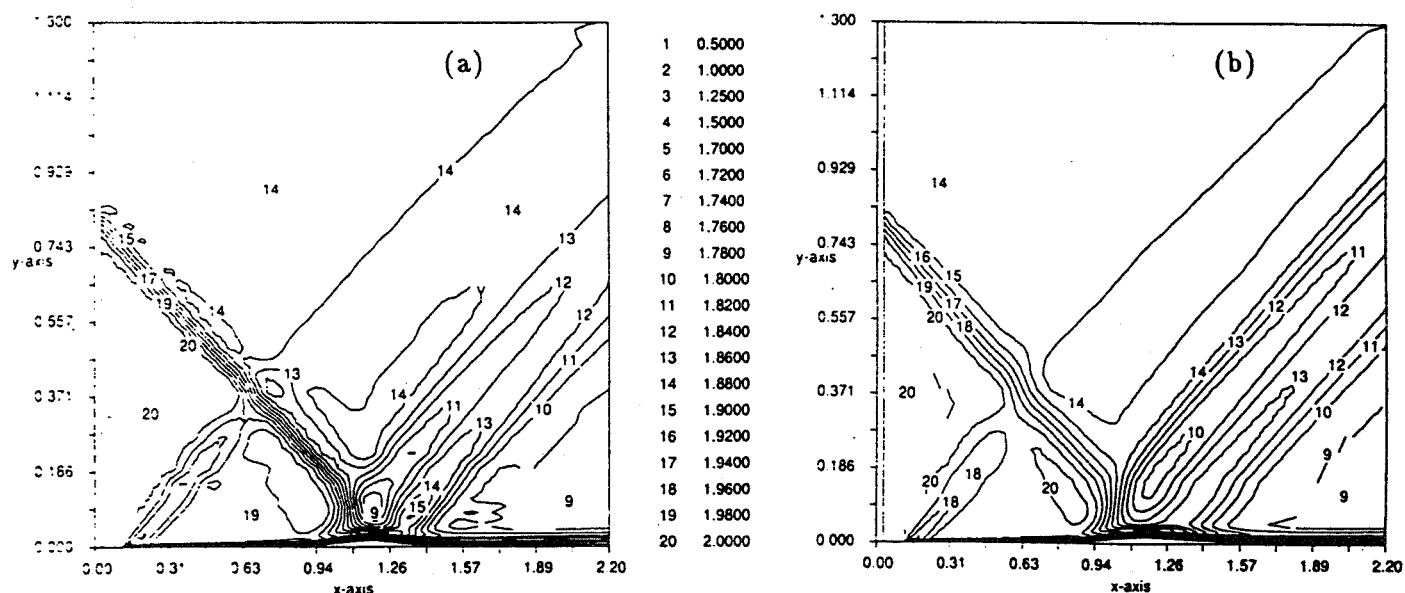


Fig. 2 Mach contours for a shock wave/boundary layer interaction problem: (a) the AUSM solution showing sharp resolutions of various waves, and (b) the Roe solution showing more diffusive waves.

Motivation

To couple *Accuracy* and *Efficiency* together within an *Upwind* Flux-Splitting Scheme

Two Possible Improvements

1. Less Diffusive **FVS** Technique

- High Order Polynomial Expansion (**HOPE**)
- Unresolved Stability Concerns

2. New *Advective Velocity* Splitting Technique

- Advection Upwind Splitting Method (**AUSM**)
- Simple, Accurate, Efficient and Robust

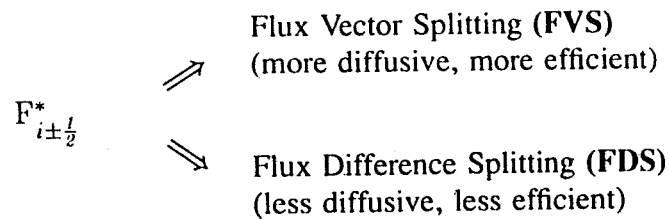
Upwinding in 1D

Conservation Law

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} = 0$$

$$u_i^{n+1} - u_i^n = -\frac{\Delta t}{\Delta x} \left(F_{i+\frac{1}{2}}^* - F_{i-\frac{1}{2}}^* \right)$$

Numerical Flux Function



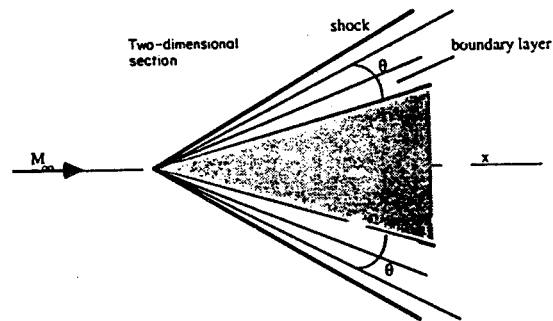
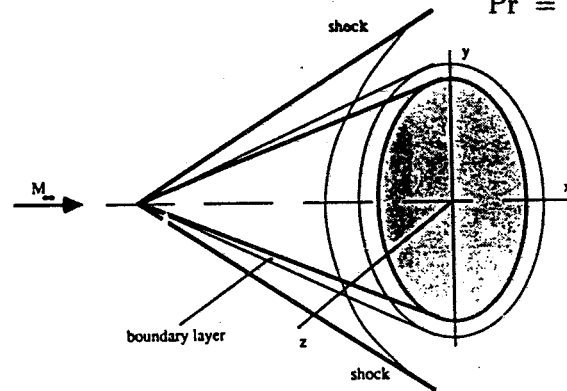
Quasi-2D Viscous Conical Flow

Geometry

$$M_{\infty} = 7.95$$

$$Re_{\infty} = 0.42 \times 10^6$$

$$Pr = 1.0$$



Adiabatic Wall Boundary Condition

AUSM in 1D

Euler Equations

$$\frac{\partial \bar{U}}{\partial t} + \frac{\partial \bar{F}}{\partial x} = 0, \quad \bar{U} = \begin{Bmatrix} \rho \\ \rho u \\ \rho E \end{Bmatrix}, \quad \bar{F} = \begin{Bmatrix} \rho u \\ \rho uu \\ \rho Hu \end{Bmatrix} + \begin{Bmatrix} 0 \\ p \\ 0 \end{Bmatrix}$$

Convective Flux

$$\bar{F}_c = (\text{scalar})(\text{convective velocity})$$

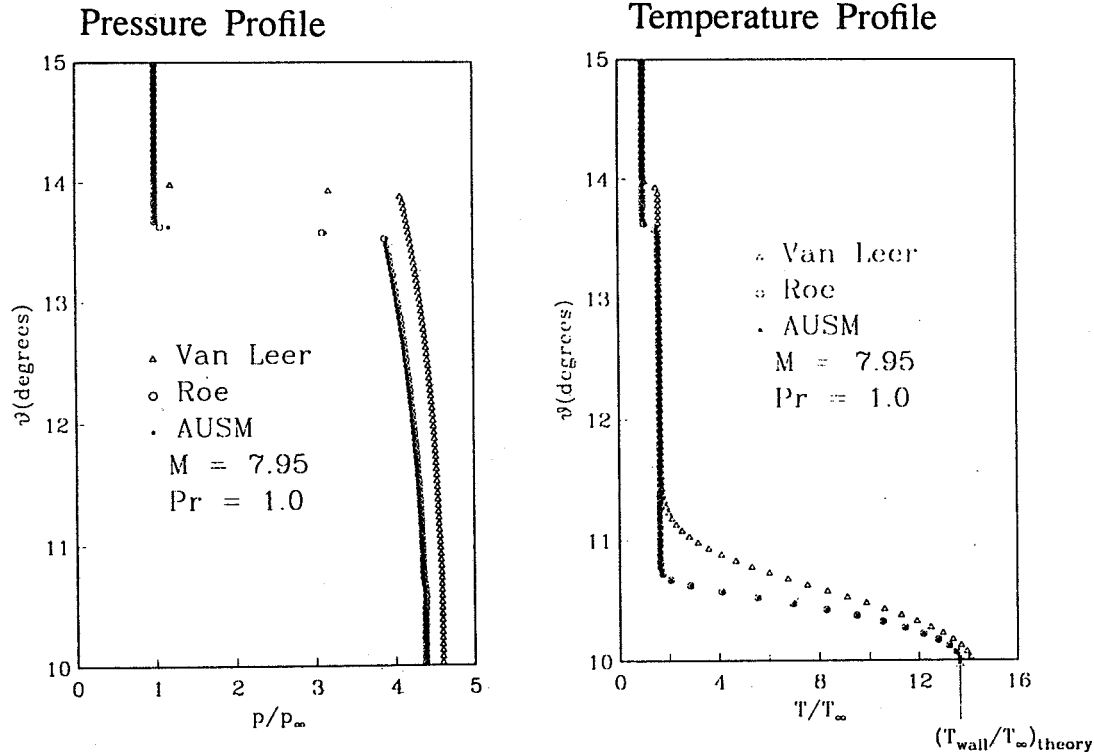
$$\bar{F}_{c_{i+\frac{1}{2}}}^* = (\text{scalar})_{upwind} (u)_{i+\frac{1}{2}}, \quad (u)_{i+\frac{1}{2}} = u(\bar{U}_i, \bar{U}_{i+1})$$

Pressure Flux

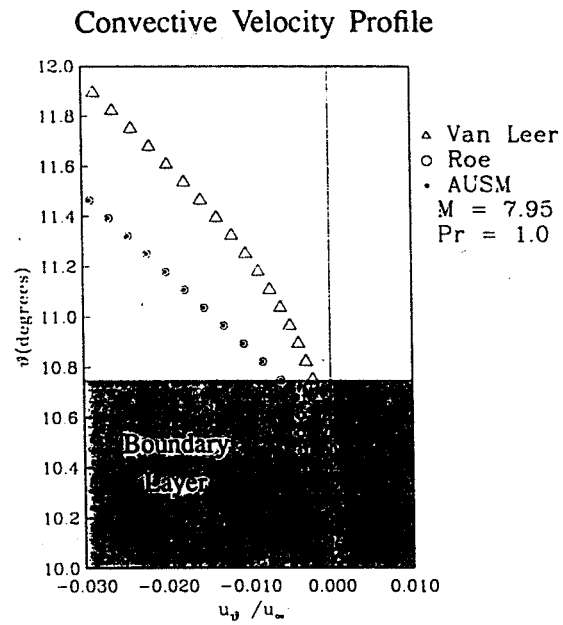
$$\bar{F}_p = (\text{scalar pressure})$$

$$\bar{F}_{p_{i+\frac{1}{2}}}^* = (p^+)_i + (p^-)_{i+1}, \quad (p^\pm)_i = p(\bar{U}_i)$$

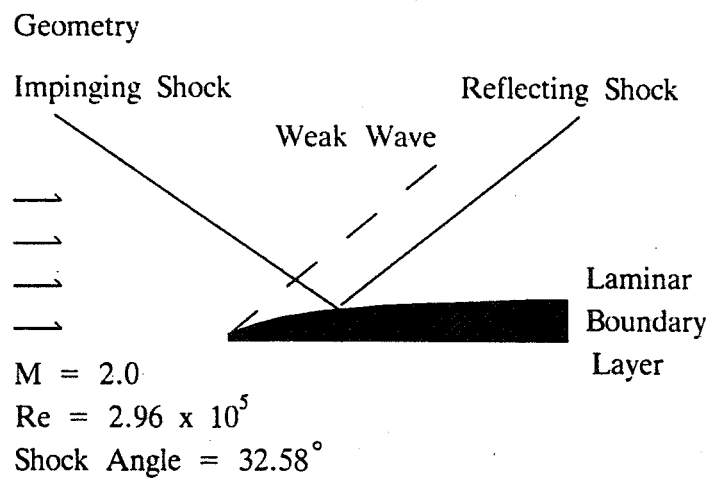
Conical Flow: First Order Accurate Results



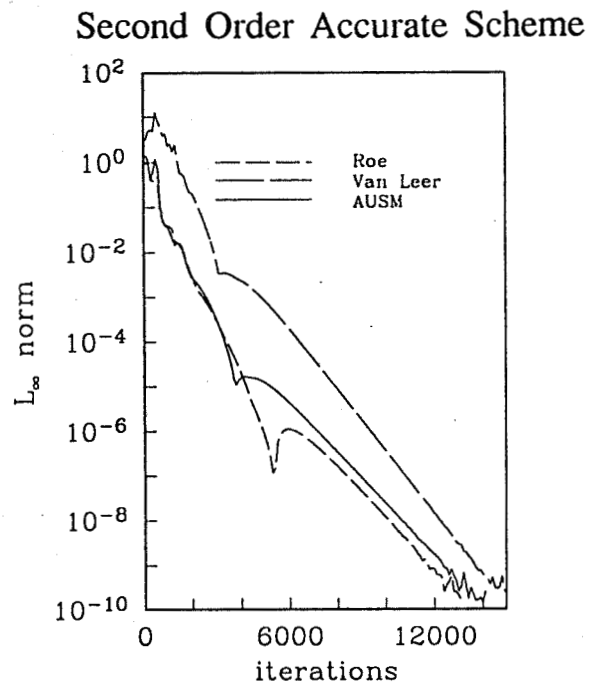
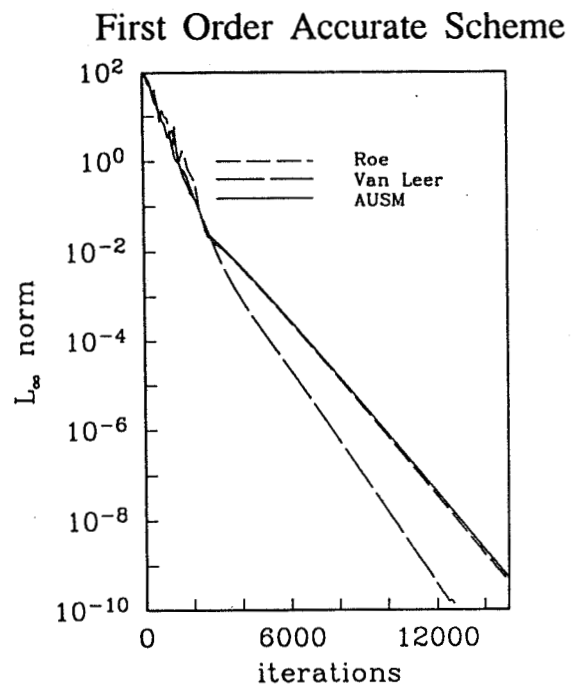
Conical Flow: First Order Accurate Results



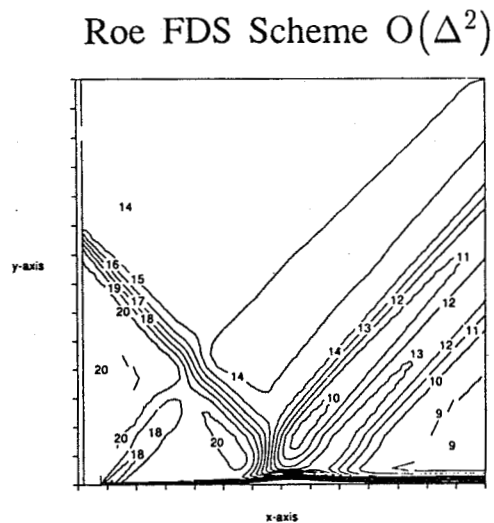
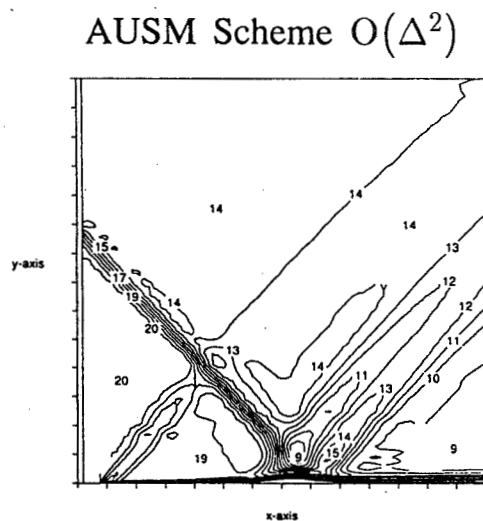
2D Shock Wave - Boundary Layer Interaction



Conical Flow: Convergence Histories

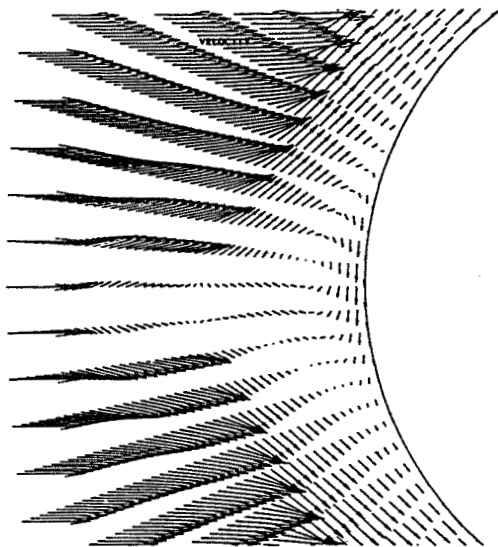


Shock Wave/Boundary Layer: Mach Number Contours



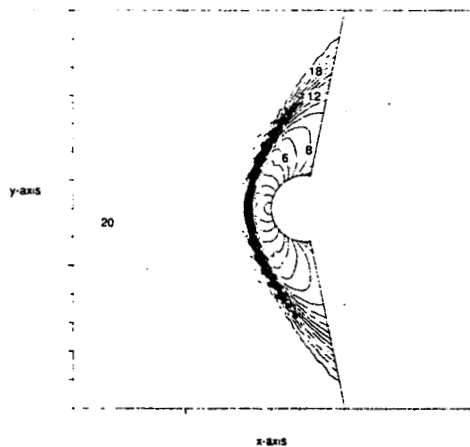
Mach 6.0 Blunt Body Flow: Carbuncle Phenomenon

Velocity Vectors

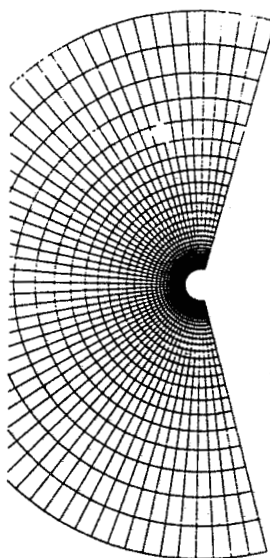


Mach 3.0 Blunt Body Flow: Mach Number Contours

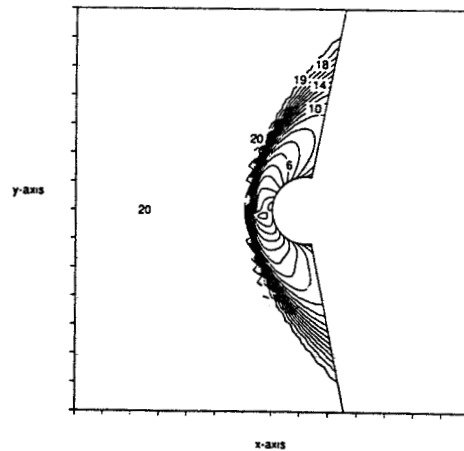
AUSM Scheme $O(\Delta)$



Grid

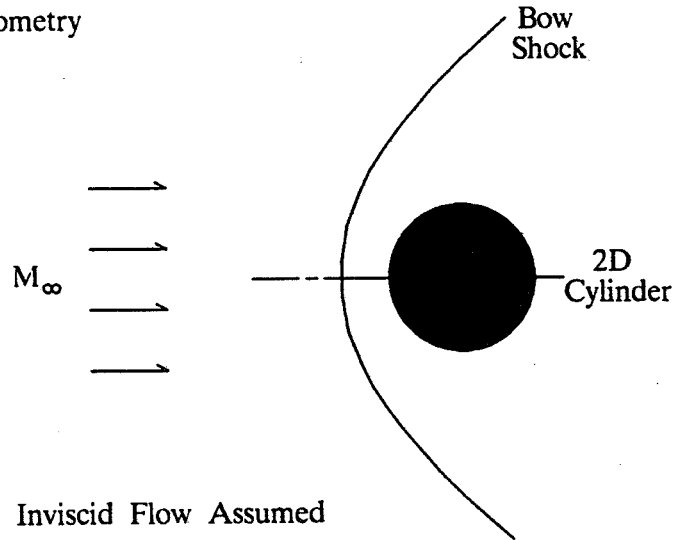


Roe FDS Scheme $O(\Delta)$



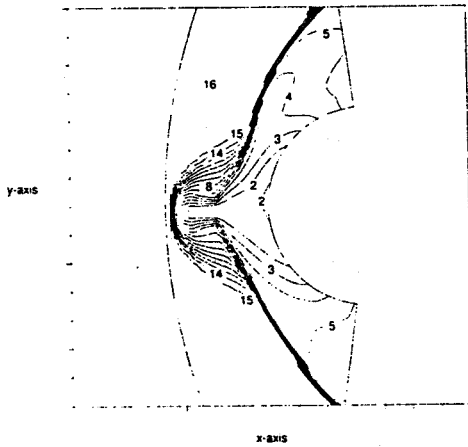
Supersonic Blunt Body Flow

Geometry

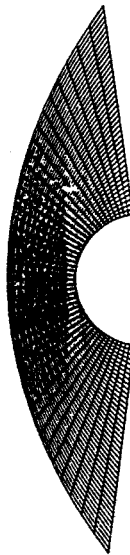


Mach 6.0 Blunt Body Flow: Mach Number Contours

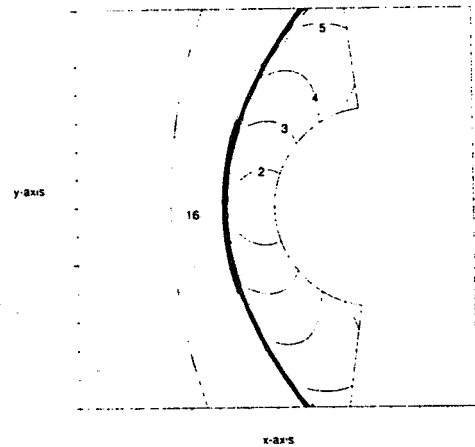
Roe FDS Scheme $O(\Delta)$



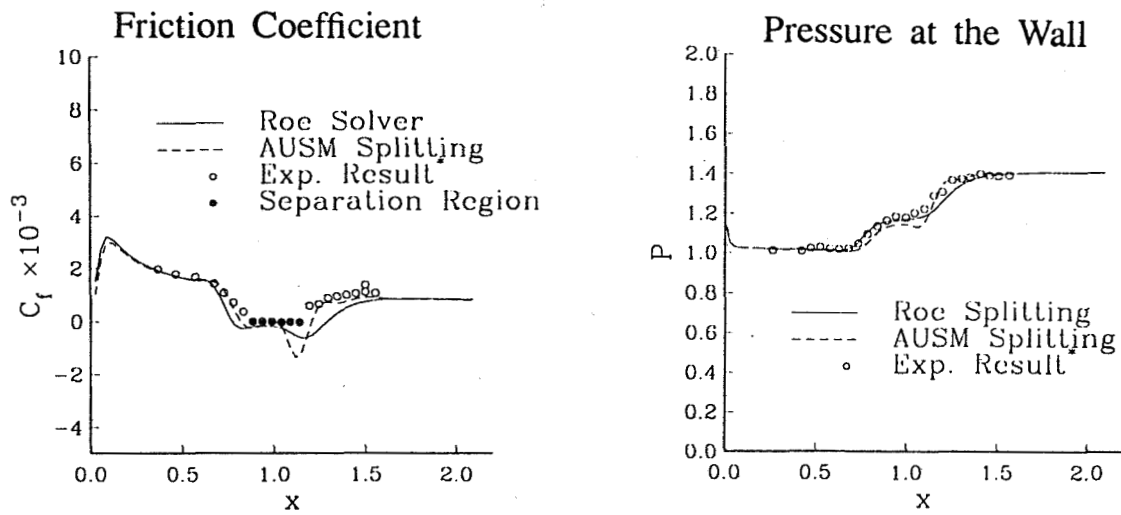
Grid



AUSM Scheme $O(\Delta)$



Shock Wave/Boundary Layer:



*Experiment by Hakkinen, Greber, Trilling and Abarbanel

Conclusions

- Accuracy of **AUSM** rivals **FDS** schemes (i.e. Roe Splitting)
- Efficiency and Simplicity of **AUSM** rivals **FVS** schemes (i.e. VanLeer Splitting)
- **AUSM** is a new family of Upwind Flux Splitting Techniques

AN ITERATIVE IMPLICIT DDADI ALGORITHM
FOR SOLVING THE NAVIER-STOKES EQUATION

S.C. CHEN, N.S. LIU, and H.D. KIM
INTERNAL FLUID MECHANICS DIVISION
NASA LEWIS RESEARCH CENTER

An algorithm utilizing a first-order upwind split flux technique and the diagonally dominant treatment is proposed to be the temporal operator for solving the Navier-Stokes equations. During the factorization process, if care is taken to ensure the symmetry of the operator, the resulting algorithm will be stable when spatial derivatives of the right hand side residual are evaluated by the central differencing scheme as well as the upwind differencing Roe scheme. Temporal accuracy is assured by the inner Newton-iteration process first introduced by Rai (1986).

Given the limit of a five-point stencil, the right hand side flux derivatives are formulated by several commonly used central and upwind schemes. Their performances are studied through a test case of free vortex convection in a uniform stream. From these results, a superior treatment for evaluating the flux term is proposed and compared with the rest.

The application of the proposed algorithm to the full Navier-Stokes equations is demonstrated through a calculation of flow over a backward facing step. Results are compared against the calculation done by using the fourth-order central differencing scheme with artificial damping.

OBJECTIVE

**GIVEN A STENCIL OF 5 POINTS, FIND A
SCHEME THAT COMBINES THE MERITS
OF BOTH CENTRAL DIFFERENCING AND
UPWIND DIFFERENCING SCHEMES.**

Temporal Operator

$$\left[\alpha I + D_{j,k} \Delta t + S_{j,k} \Delta t + L_{j-1,k-1} \Delta t + U_{j+1,k+1} \Delta t \right] \Delta Q^{(P)}$$

$$= \Delta t \cdot \text{RHS}$$

$$\begin{aligned} \text{RHS} = & -\alpha \sum_{q=1}^{P-1} \Delta Q^{(q)} / \Delta t + (\alpha - 1) \Delta Q^{(n-1)} / \Delta t \\ & - \frac{\partial E}{\partial \xi} - \frac{\partial F}{\partial \eta} - \frac{\partial G}{\partial \zeta} + \frac{\partial(Ev)}{\partial \xi} + \frac{\partial(Fv)}{\partial \eta} + \frac{\partial(Gv)}{\partial \zeta} \end{aligned}$$

Where $D = A^+ - A^- + B^+ - B^- + C^+ - C^-$
 $S = \partial(\text{source terms}) / \partial Q$
 L - Lower half of the flux Jacobian operators
 U - Upper half of the flux Jacobian operators

Algorithm: In 2 Consecutive Steps

$$[\alpha I + D \Delta t + S \Delta t + (L \Delta t)] \Delta Q^{(n)} = \Delta t \cdot \text{RHS}^{(n)}$$

$$\begin{aligned} & [\alpha I + D \Delta t + S \Delta t] \Delta Q^{(n)} + [\Delta t L(\Delta Q^{(n)}) + \Delta t U(\Delta Q^{(n)})] \\ & = \Delta t \cdot \text{RHS}^{(n)} \end{aligned}$$

$$[\alpha I + D \Delta t + S \Delta t + (U \Delta t)] \Delta Q^{(n+1)} = \Delta t \cdot \text{RHS}^{(n+1)}$$

$$\begin{aligned} & [\alpha I + D \Delta t + S \Delta t] \Delta Q^{(n+1)} + [\Delta t L(\Delta Q^{(n+1)}) + \Delta t U(\Delta Q^{(n+1)})] \\ & = \Delta t \cdot \text{RHS}^{(n+1)} \end{aligned}$$

∴ Formal Error in 2 Time Steps is

$$2 \cdot E_{RR} \sim -\Delta t^2 \cdot U \left(\frac{\delta \Delta Q^*}{\delta t} \right)$$

$$+\Delta t^2 \cdot \left\{ L \left[\alpha I + \Delta t(D+S) \right]^{-1} U \right\} (\Delta Q^*)$$

$$-\Delta t^2 \cdot L \left(\frac{\delta \Delta Q^{**}}{\delta t} \right)$$

$$+\Delta t^2 \cdot \left\{ U \left[\alpha I + \Delta t(D+S) \right]^{-1} L \right\} (\Delta Q^{**})$$

Define

$$\frac{\delta Q}{\delta t} \equiv \alpha \frac{\Delta Q^{(n)}}{\Delta t} - (\alpha-1) \frac{\Delta Q^{(n-1)}}{\Delta t}$$

$$= \alpha \sum_{q=1}^P \frac{\Delta Q^{(q)}}{\Delta t} - (\alpha-1) \frac{\Delta Q^{(n-1)}}{\Delta t}$$

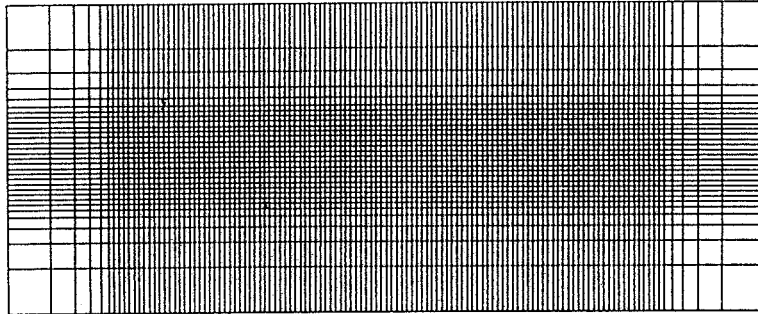
Then

$$RHS = -\frac{\delta Q}{\delta t} - \alpha \frac{\Delta Q^{(P)}}{\Delta t}$$

$$-\frac{\delta E}{\delta \xi} - \frac{\delta F}{\delta \eta} - \frac{\delta G}{\delta \zeta} + \frac{\delta(Ev)}{\delta \xi} + \frac{\delta(Fv)}{\delta \eta} + \frac{\delta(Gv)}{\delta \zeta}$$

Geometry

$$\left\{ \begin{array}{l} \Delta x = \Delta y = 1/4 \quad (\text{Vortex core radius} = 1) \\ \Delta t = 0.05 \\ \text{CFL} \sim 0.7 \\ M_\infty = 0.5 \\ \Gamma = -0.5 \\ \text{Innerm} = 4; \text{NORDR} = 2 \\ \text{Invisid} \end{array} \right. \quad \left\{ \begin{array}{l} P_\gamma = 0.72 \\ \gamma = 1.4 \\ T_\infty = 519^\circ \text{R} \\ \rho_\infty = 2.374 \times 10^{-3} \text{ slug/ft}^3 \\ \text{Air} \end{array} \right.$$



$$U, V, P, T = f_n(t)$$

Travel a total of 45 length in 900 steps.

Test 1 4th Order Central

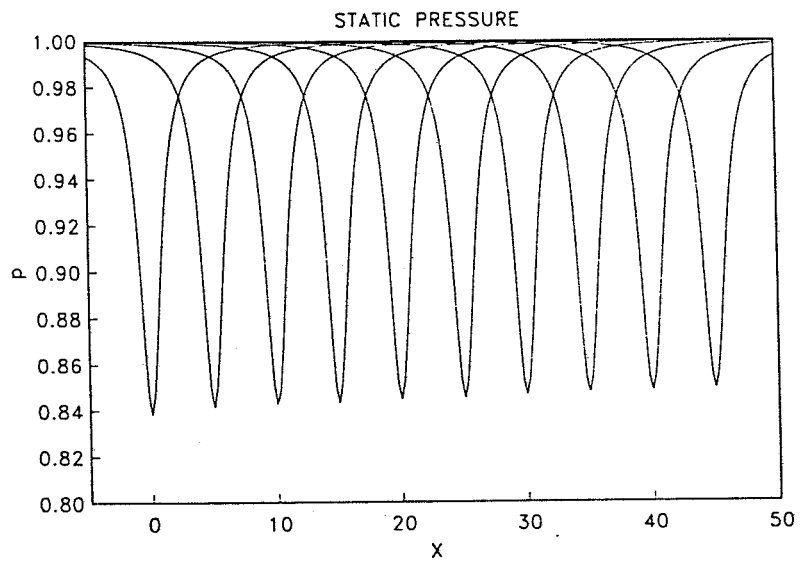
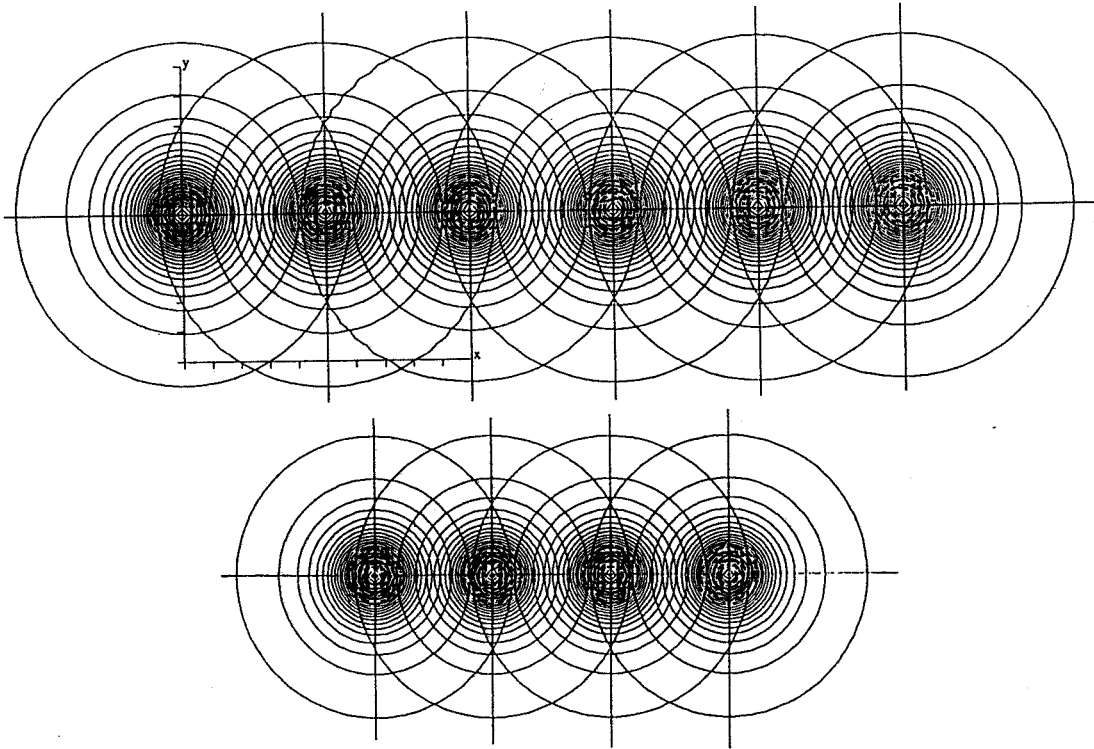
Flux derivative in 1-D difference for

$$E'_i = \left[\frac{2}{3h} (E_{i+1} - E_{i-1}) - \frac{1}{12h} (E_{i+2} - E_{i-2}) \right]$$

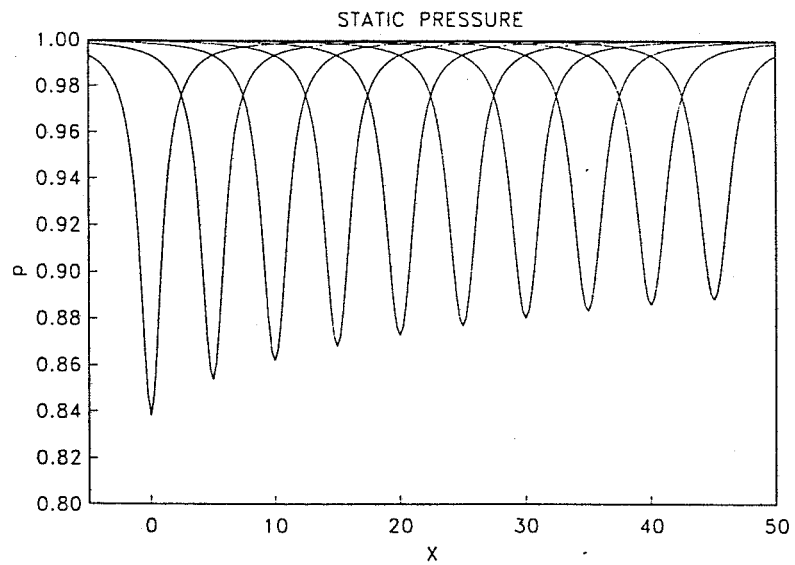
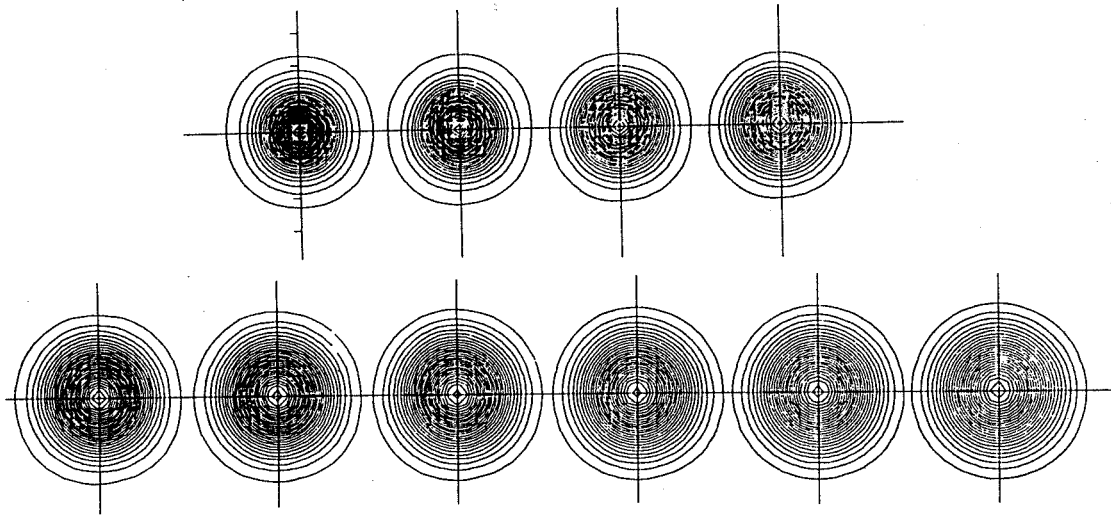
$$- \sum_{n=5}^{N,2} \frac{1}{3} \frac{(4-2^{n-1})}{n!} E_i^{(n)} h^{n-1} (N \rightarrow \infty)$$

* Contains only the odd derivatives leading error is of order "4"

Pressure



Vorticity Magnitude



Consider:

$E'_i = [4\text{th order central differencing}]$

$$-\sum_{n=5}^{N,2} \frac{1}{3} \frac{(4-2^{n-1})}{n!} E_i^{(n)} h^{n-1} \quad (A)$$

and

$E'_i = [3\text{rd order upwind biased differencing}]$

$$-\sum_{n=5}^{N,2} \frac{1}{3} \frac{(4-2^{n-1})}{n!} E_i^{(n)} h^{n-1}$$

$$-\sum_{n=4}^{N,2} \frac{1}{3} \frac{(2-2^{n-1})}{n!} E_i^{(n)} h^{n-1} \quad (B)$$

The Dissipative Equivalence is

$$\text{Diss} = (A) - (B)$$

i.e.,

$$\text{Diss} = -\sum_{n=4}^{N,2} \frac{1}{3} \frac{(2-2^{n-1})}{n!} E_i^{(n)} h^{n-1}$$

= [3rd order upwind difference]

- [4th order central difference]

Let $\frac{\delta E}{\delta x} = 4\text{th central} + \gamma \cdot \text{DISS}$

Then =

$$\begin{aligned}
 E_i' &= \{4\text{th order central differencing} \\
 &+ \gamma \cdot [-\sum_{n=4}^{N,2} \frac{1}{3} \frac{(2-2^{n-1})}{n!} E_i^{(n)} h^{n-1}]\} \\
 &- \sum_{n=5}^{N,2} \frac{1}{3} \frac{(4-2^{n-1})}{n!} E_i^{(n)} h^{n-1} \\
 &= \{(1-\gamma) \cdot [4\text{th order central difference}] \\
 &+ \gamma \cdot [3\text{rd order upwind biased difference}]\} \\
 &+ \text{Trunc. error}
 \end{aligned}$$

Test 5. 3rd Order Upwind Biased Scheme

$$E_i' = \left[\frac{1}{3h} (E_{i+1} - E_{i-1}) + \frac{1}{6h} (-3 E_i + 4 E_{i+1} - E_{i+2}) \right]$$

$$-\sum_{n=5}^{N,2} \frac{1}{3} \frac{(4-2^{n-1})}{n!} E_i^{(n)} h^{n-1} \quad (\text{odd derivatives})$$

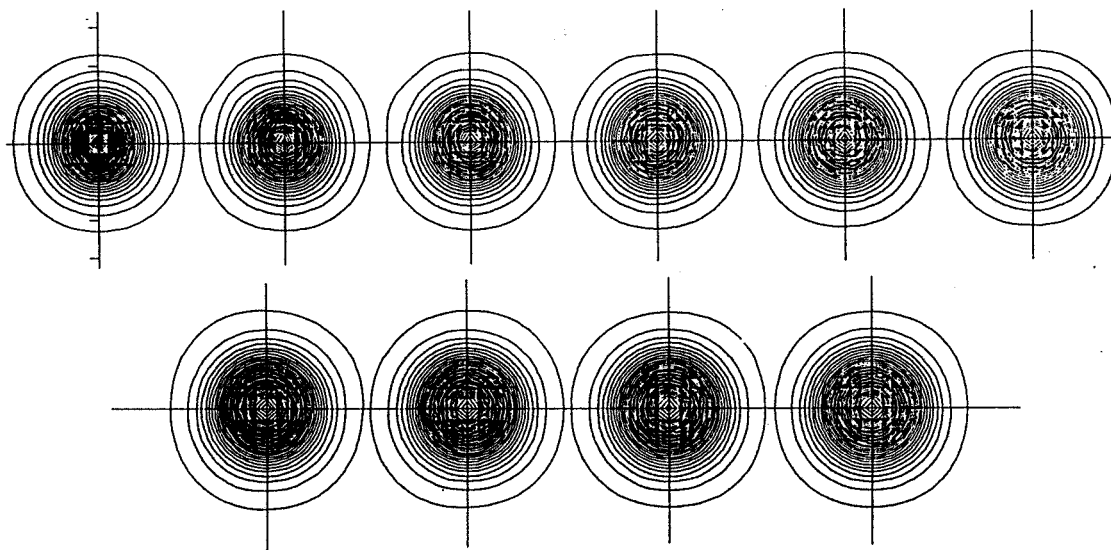
$$-\sum_{n=4}^{N,2} \frac{1}{3} \frac{(2-2^{n-1})}{n!} E_i^{(n)} h^{n-1} \quad (\text{even derivatives})$$

* Leading error is of order "3"

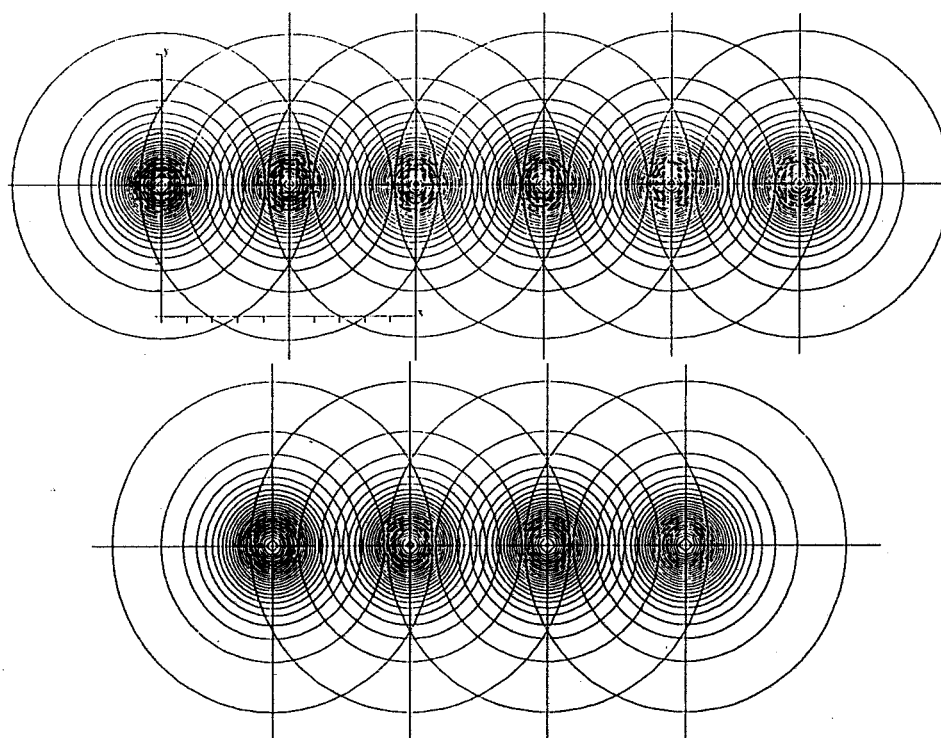
Test 6

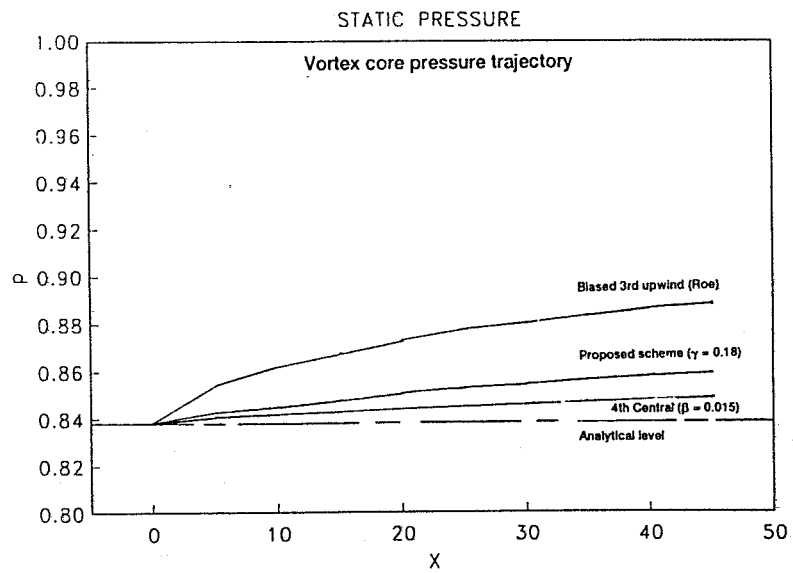
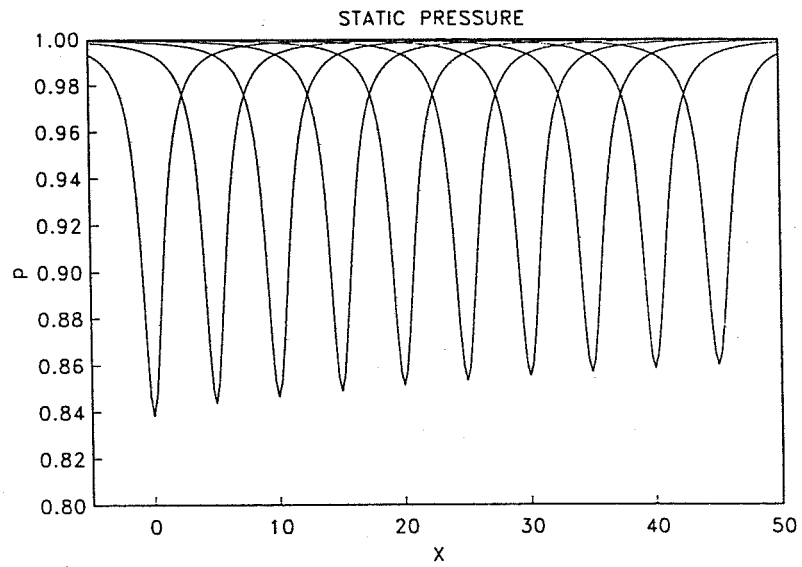
Set $\gamma = 0.18$ so that leading error term has a coefficient of 0.015

Vorticity Magnitude



Pressure





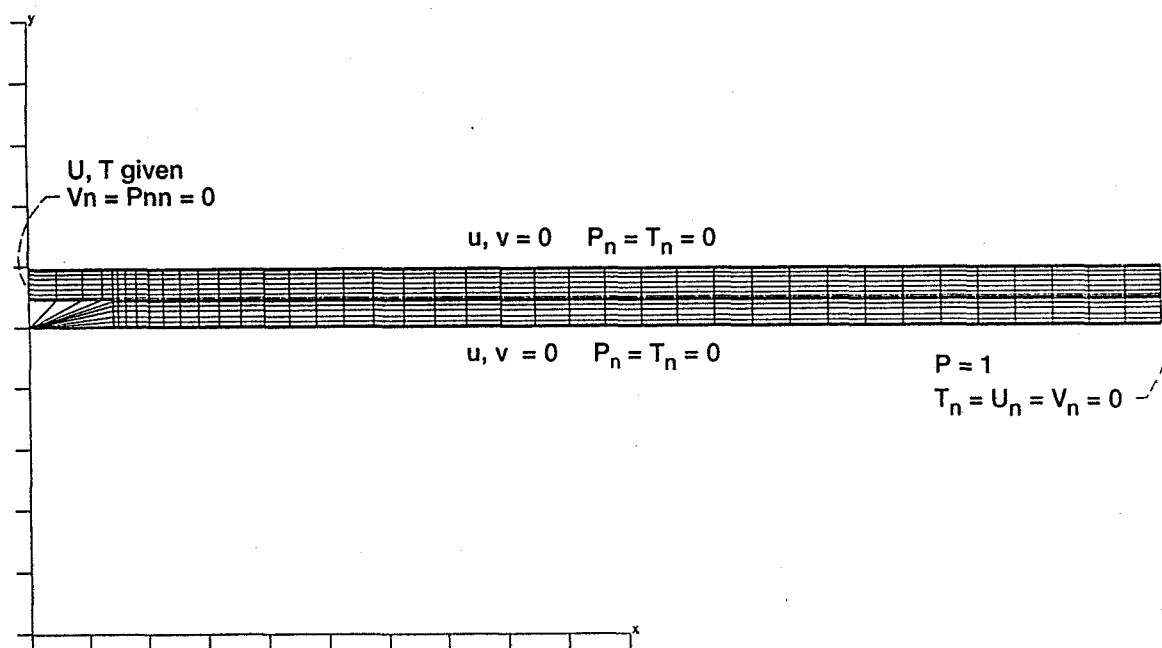
Case 2

Flow over a backward-facing step

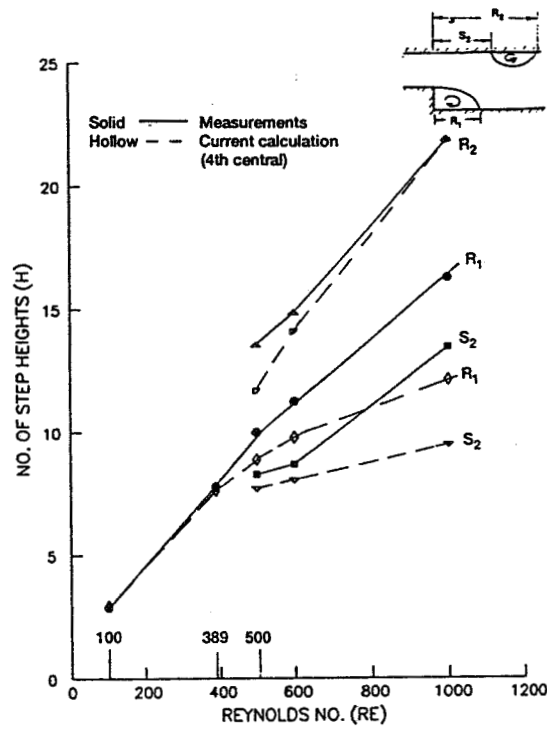
Re_d	100	100
	389	389
	500	1000
	600	
	1000	
	↑	↑
	4th order central + artificial viscosity ($\beta = 0.5$)	Proposed scheme ($\gamma = 0.5$)

Max. L_2 - Residual $< 5 \times 10^{-5}$ for all cases

$$\left\{ \begin{array}{ll} \overline{M}_\infty = 0.18 & CFL = 50.0 \\ Pr = 0.72 & NORDR = 1 \\ \gamma = 1.4 & Innerm = 1 \\ \text{Air; viscous flow} & \end{array} \right.$$

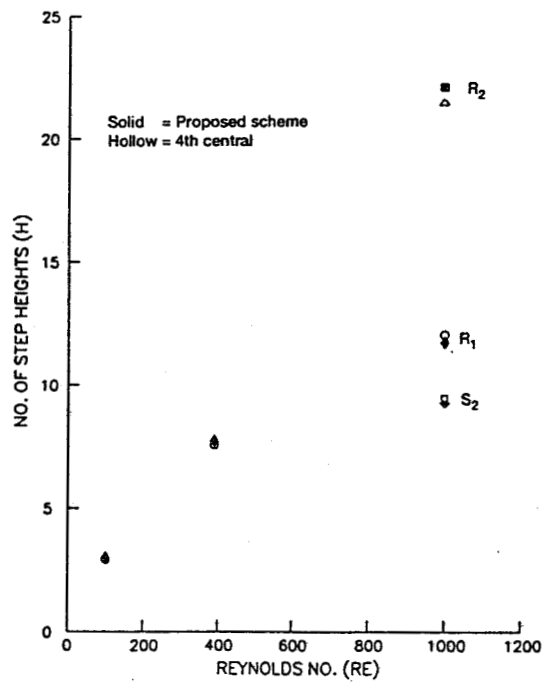


Reattachment and Detachment Length I



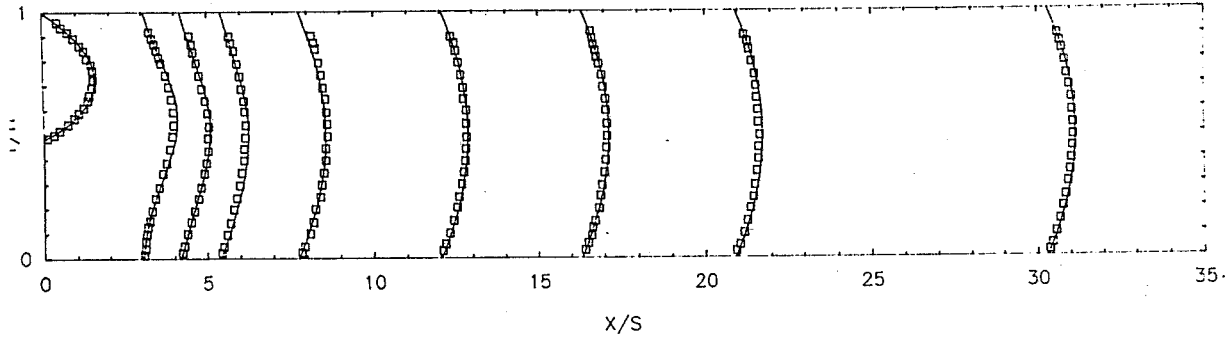
Ref. - JFM: vol 127, pp. 473-496. Armaly et al.

Reattachment and Detachment Length II



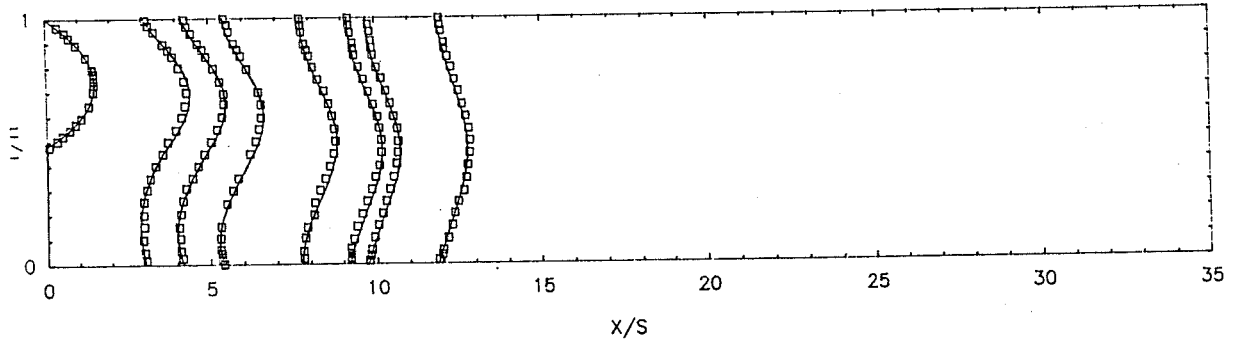
□ Measurements
 — Proposed scheme
 - - - 4th Order central

V - Velocity profiles
Re = 100

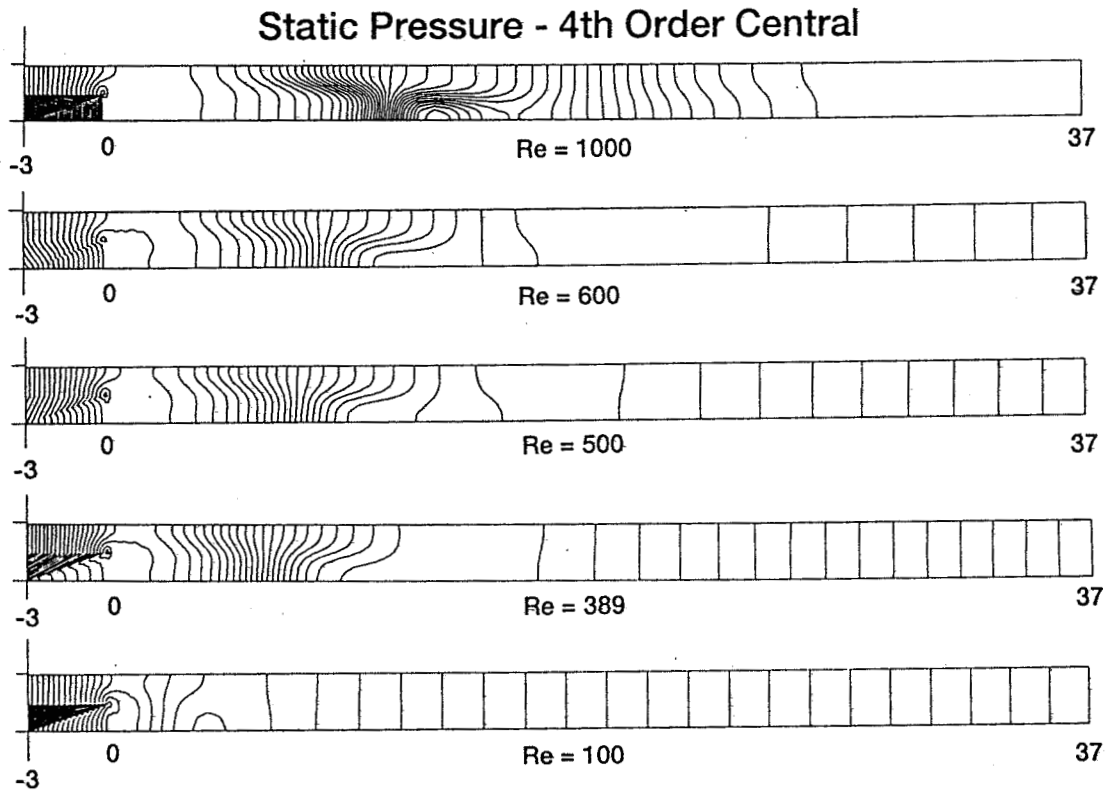
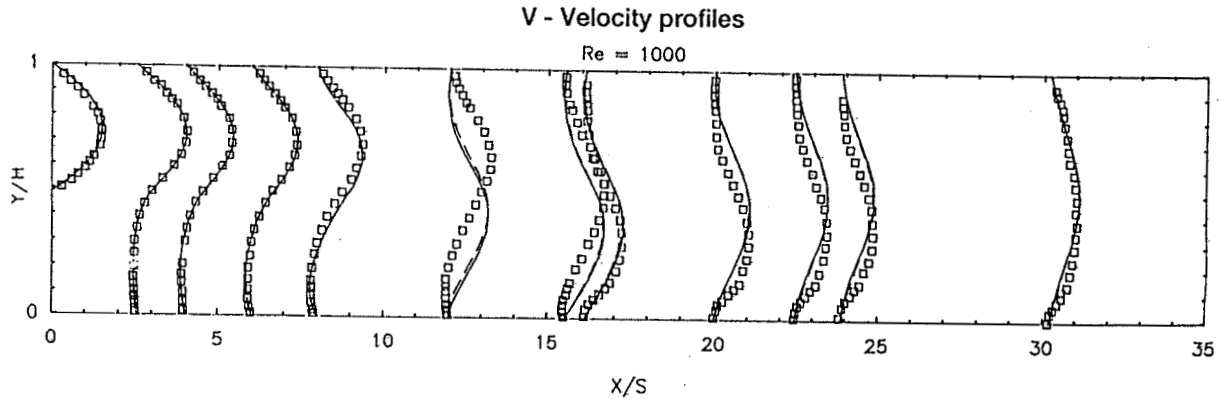


□ Measurements
 — Proposed scheme
 - - - 4th Order central

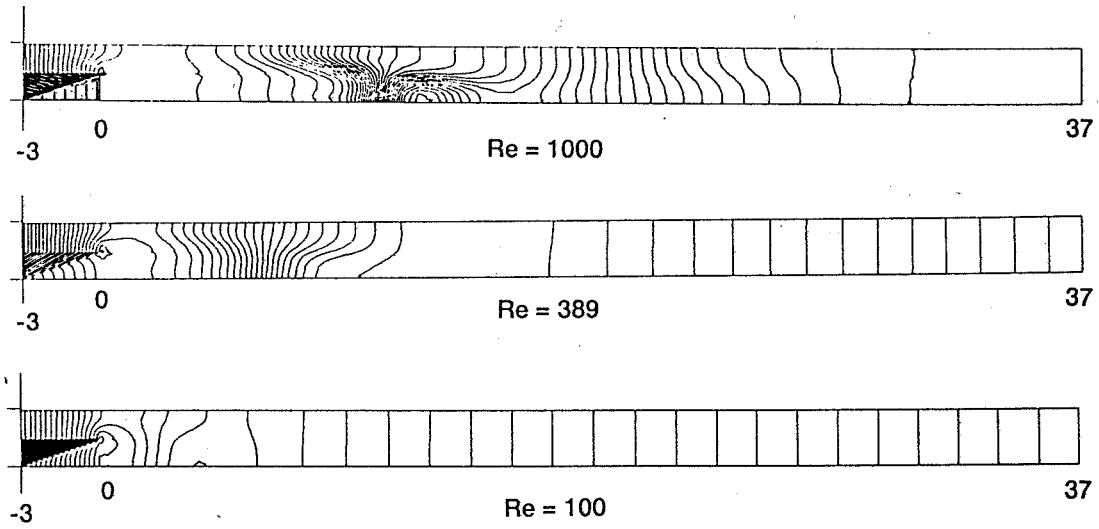
V - Velocity profiles
Re = 389



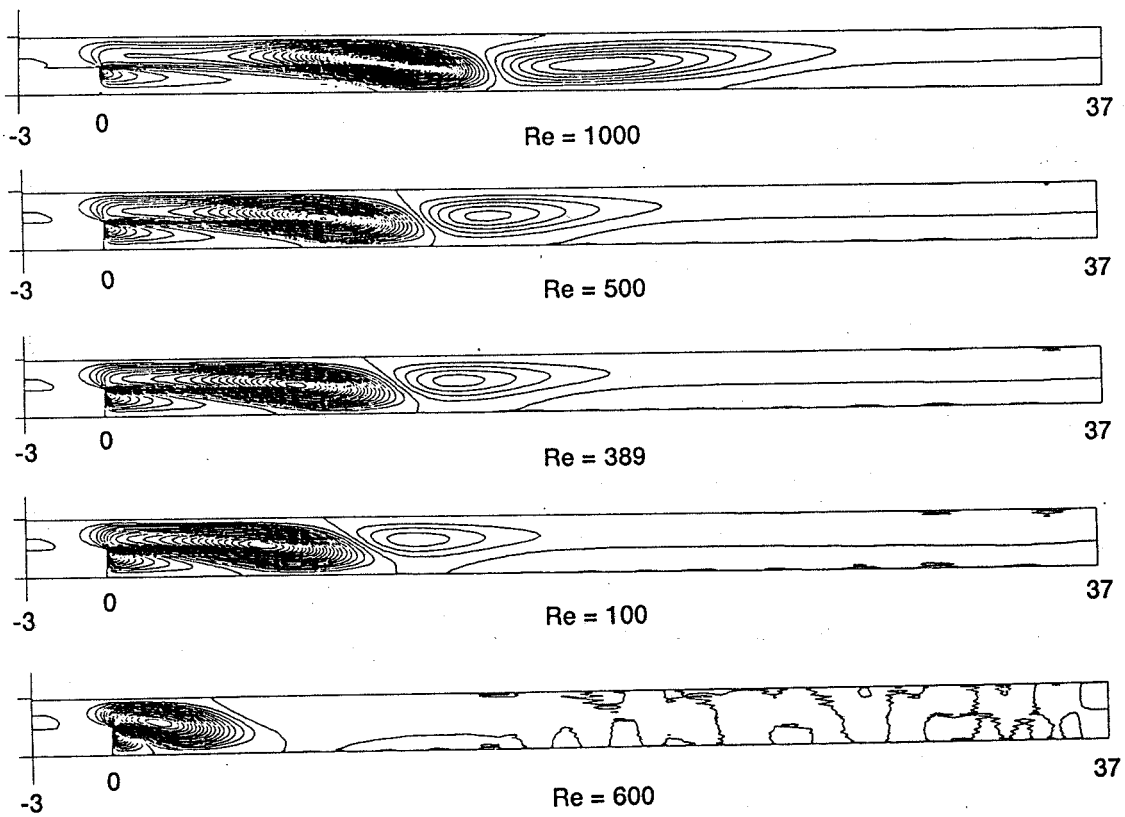
- Measurements
- Proposed scheme
- - - 4th Order control



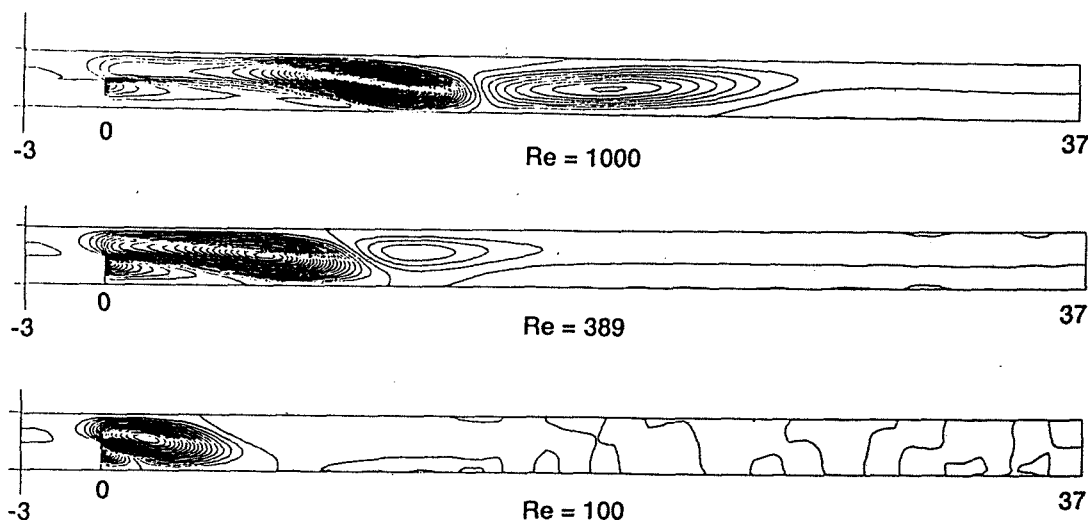
Static Pressure - Proposed Scheme



V Component - 4th Order Central



V Component - Proposed Scheme



LEWIS RESEARCH CENTER

CONCLUDING REMARKS

*** CENTRAL DIFFERENCING:**

SUSCEPTIBLE TO SPURIOUS OSCILLATIONS,
ARTIFICIAL DAMPING NEEDS TO BE ADDED.

*** UPWIND DIFFERENCING:**

INTRINSIC YET EXCESSIVE NUMERICAL DISSIPATION.

*** PROPOSED SCHEME:**

USES 4th ORDER CENTRAL DIFFERENCING SCHEME WITH
INTRINSIC NUMERICAL DISSIPATION PROVIDED BY THE
UPWIND DIFFERENCING SCHEME.

INTERNAL FLUID MECHANICS DIVISION

The Proteus Navier-Stokes Code

Charles E. Towne and John R. Schwab
Internal Fluid Mechanics Division
NASA Lewis Research Center

An effort is currently underway at NASA Lewis to develop two- and three-dimensional Navier-Stokes codes, called Proteus, for aerospace propulsion applications. The emphasis in the development of Proteus is not algorithm development or research on numerical methods, but rather on the development of the code itself. The objective is to develop codes that are user-oriented, easily-modified, and well-documented. Well-proven, state-of-the-art solution algorithms are being used. Code readability, documentation (both internal and external), and validation are being emphasized.

Proteus solves the Reynolds-averaged, unsteady, compressible Navier-Stokes equations in strong conservation law form. Turbulence is modeled using a Baldwin-Lomax based algebraic eddy viscosity model (AIAA Paper 78-257.) The governing equations are written in Cartesian coordinates and transformed into generalized nonorthogonal body-fitted coordinates. They are solved by marching in time using a fully-coupled ADI solution procedure (Briley and McDonald, J. Comp. Phys., Aug. 1977) with generalized first- or second-order time differencing (Beam and Warming, AIAA J., Apr. 1978.) The boundary conditions are also treated implicitly, and may be steady or unsteady. All terms, including the diffusion terms, are linearized using second order Taylor series expansions.

In addition to solving the full set of Reynolds-averaged equations, options are available to solve the thin-layer or Euler equations, and to eliminate the energy equation by assuming constant stagnation enthalpy. Artificial viscosity is used to minimize the odd-even decoupling resulting from the use of central spatial differencing for the convective terms, and to control pre- and post-shock oscillations in supersonic flow. Two artificial viscosity models are available - a combination implicit/explicit constant coefficient model (Steger, AIAA J., July 1978), and an explicit nonlinear coefficient model designed specifically for flows with shock waves (Jameson, Schmidt, and Turkel, AIAA Paper 81-1259.) At NASA Lewis, the code is run on the Cray X-MP and Y-MP computers, and is highly vectorized.

An extensive series of validation cases have been run, primarily using the two-dimensional planar/axisymmetric version of the code. Several flows were computed for which exact solutions to the Navier-Stokes equations exist, including fully-developed channel and pipe flow, Couette flow with and without a pressure gradient, unsteady Couette flow formation, flow near a suddenly accelerated flat plate, flow between concentric rotating cylinders, and flow near a rotating disk. Additional validation cases that have been successfully run include flat plate laminar and turbulent boundary layers, boundary layers with confined separation, 2-D and 3-D driven cavities, normal and oblique shock-boundary layer interactions, steady and unsteady flows past a circular cylinder, developing laminar and turbulent pipe flows, and steady and unsteady flows in a transonic diffuser. The figure shows computed Mach number contours and the static pressure distribution for flow through a transonic diffuser.

The two-dimensional version of Proteus has recently been released, and the three-dimensional code is scheduled for release in late 1991. The documentation consists of a three-volume user's manual (Towne, Schwab, Benson, and Suresh, NASA TM's 102551-102553). Volume 1 is the Analysis Description, and presents the equations and solution procedure. It describes in detail the governing equations, the turbulence model, the linearization of the equations and boundary conditions, the time and space differencing formulas, the ADI solution procedure, and the artificial viscosity models. Volume 2 is the User's Guide, and contains information needed to run the program. It describes the program's general features, the input and output, the procedure for setting up initial conditions, the computer resource requirements, the diagnostic messages that may be generated, the job control language used to run the program, and several test cases. Volume 3 is the Programmer's Reference, and contains detailed information useful when modifying the program. It describes the program structure, the Fortran variables stored in common blocks, and the details of each subprogram.

THE PROTEUS NAVIER-STOKES CODE

By

C. TOWNE

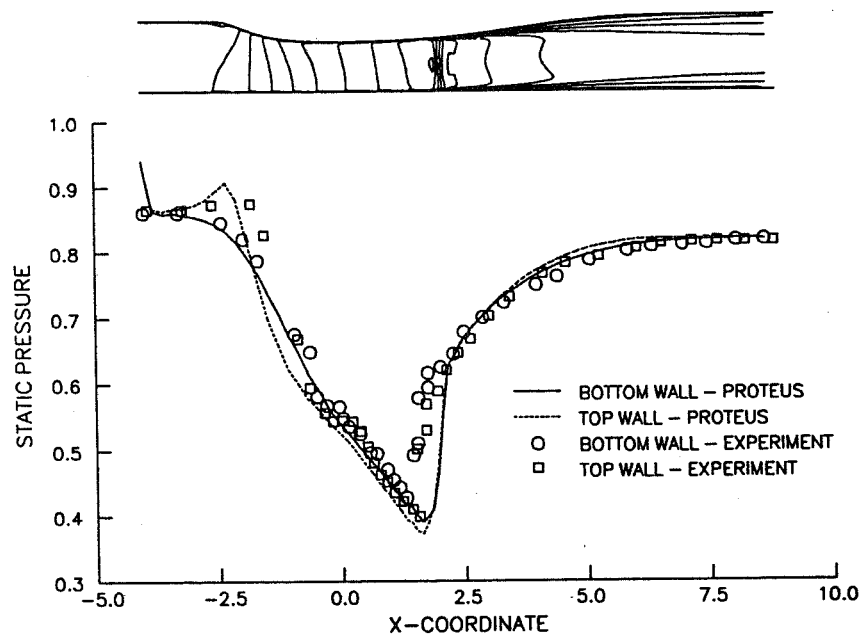
Inlet, Duct & Nozzle Flow Physics Branch

And

J. SCHWAB

Turbomachinery Flow Physics Branch

INTERNAL FLUID MECHANICS DIVISION

NAVIER-STOKES ANALYSIS
FOR TRANSONIC DIFFUSER

PROTEUS NAVIER-STOKES CODE

- Objective
 - Develop user-oriented, easily-modified, well-documented, 2- and 3-dimensional Navier-Stokes codes for aerospace propulsion applications.
- Approach
 - Use well-proven, state-of-the-art algorithms.
 - Use a consistent, modular code structure.
 - Emphasize documentation, code readability, and validation.

PROTEUS NAVIER-STOKES CODE ANALYSIS SUMMARY

- Reynolds-averaged, unsteady, compressible Navier-Stokes equations.
- Strong conservation-law form.
- Generalized nonorthogonal body-fitted coordinates.
- Convection and diffusion terms linearized using second-order Taylor series expansion.
- Baldwin-Lomax algebraic turbulence model.
- Fully-coupled ADI solution procedure, Beam-Warming generalized time differencing.
- Implicit steady/unsteady boundary conditions.

PROTEUS NAVIER-STOKES CODE

CODE FEATURES

- 2-D, axisymmetric with swirl, or 3-D flow.
- Thin-layer and Euler options.
- Wide variety of boundary conditions.
- Constant stagnation enthalpy option.
- Constant-coefficient or adaptive artificial viscosity.
- First- or second-order time differencing.
- Variety of time step selection methods.
- Output files for CONTOUR and PLOT3D.
- Highly vectorized for Cray computers.
- Extensively commented.
- Three-volume documentation set.
 - Analysis Description.
 - User's Guide.
 - Programmer's Reference.

PROTEUS NAVIER-STOKES CODE

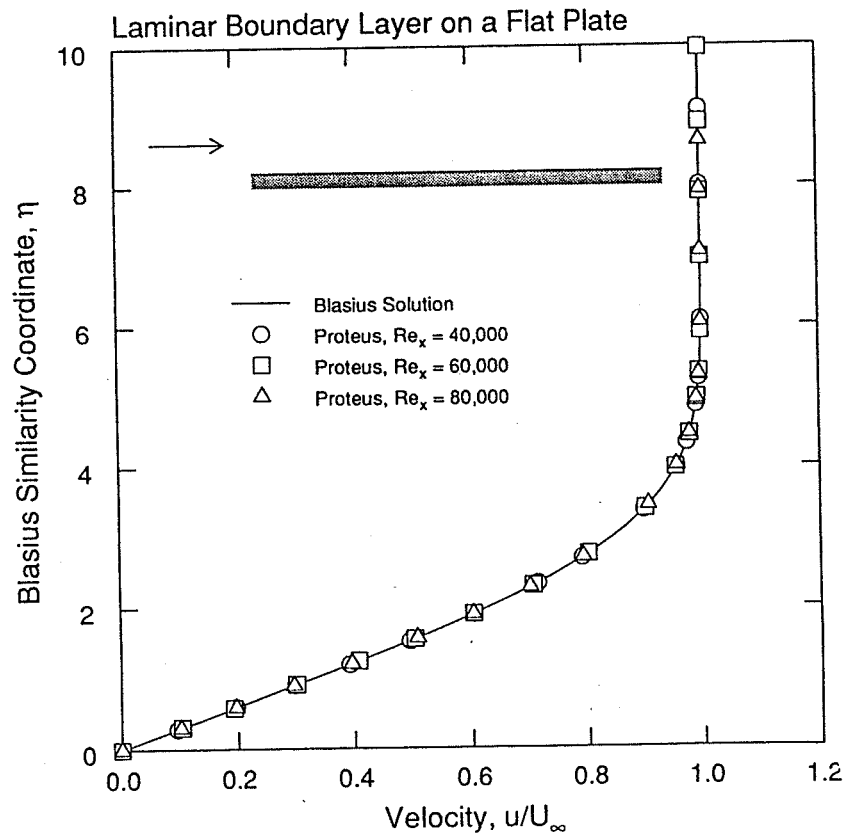
VALIDATION CASES

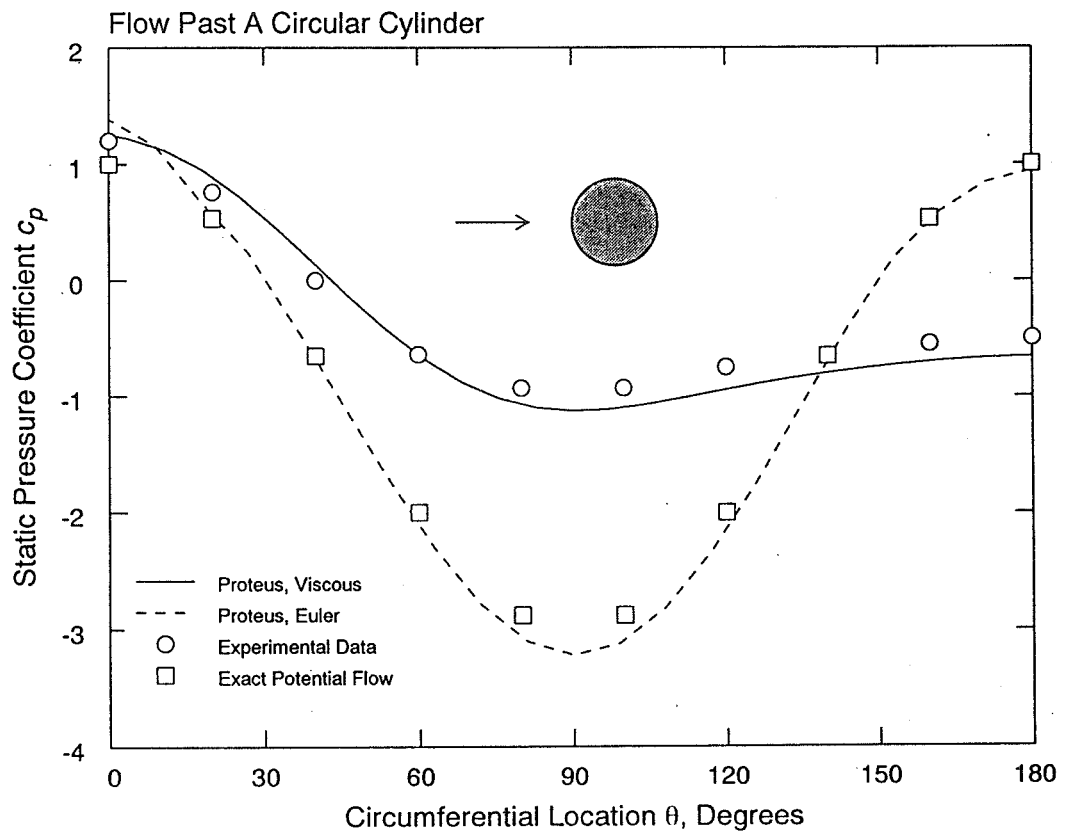
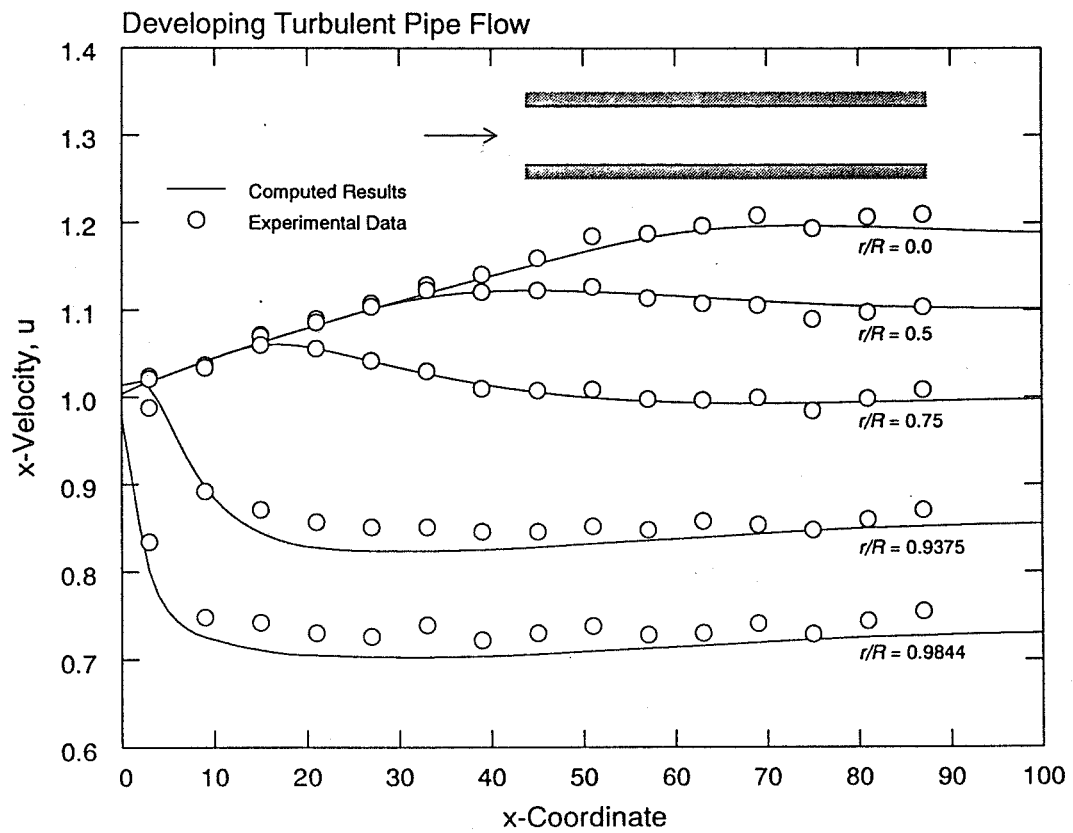
- Exact Navier-Stokes solutions.
 - Fully-developed channel and pipe flow.
 - Couette flow w/wo pressure gradient.
 - Couette flow formation.
 - Flow near a suddenly-accelerated flat plate.
 - Flow between concentric rotating cylinders.
 - Flow near a rotating disk.

PROTEUS NAVIER-STOKES CODE

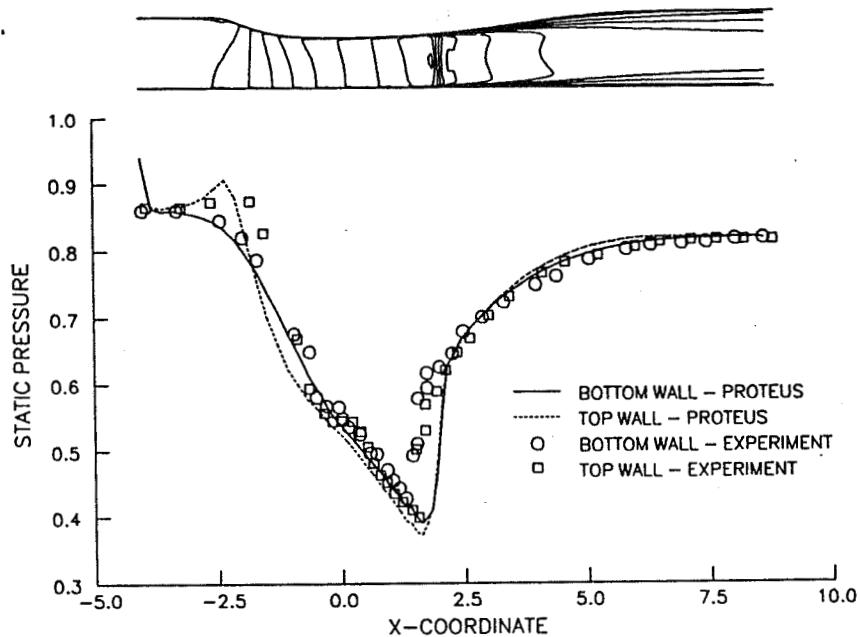
VALIDATION CASES

- Blasius flat plate boundary layer.
- Flat plate boundary layer with confined separation.
- Turbulent flat plate boundary layer w/wo heat transfer.
- Flow over a rearward-facing step.
- 2-D and 3-D driven cavity.
- Normal and oblique shock formation.
- Steady and unsteady flow past a circular cylinder.
- Flow past an airfoil.
- Developing laminar and turbulent flow in a channel and pipe w/wo swirl.
- Developing 3-D flow in a square duct.
- Steady and unsteady turbulent flow in a transonic diffuser.





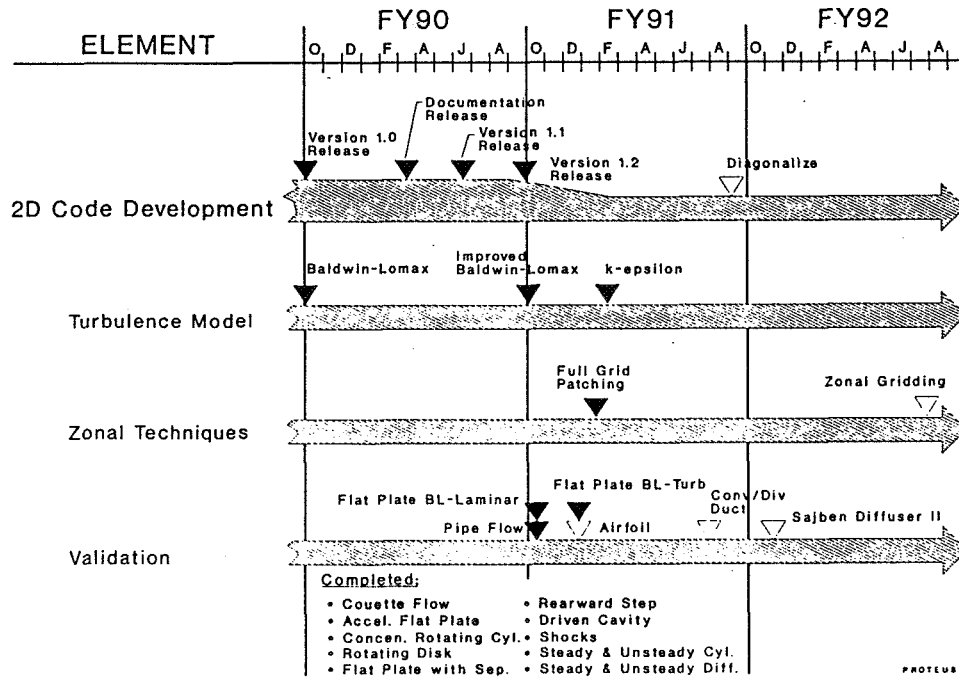
NAVIER-STOKES ANALYSIS FOR TRANSONIC DIFFUSER



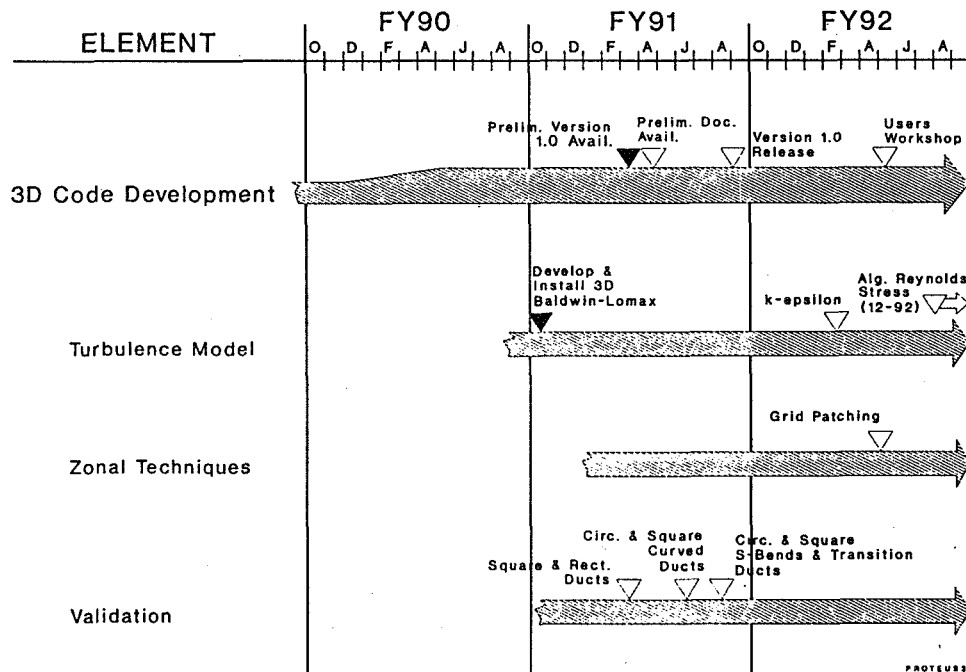
PROTEUS NAVIER-STOKES CODE HISTORY

- 4-86 Start.
- 9-87 First master file.
- 9-88 Emphasis placed on 2-D code development.
- 9-89 Version 1.0 of 2-D code released.
- 3-90 Documentation published for version 1.0 of 2-D code.
- 9-90 Version 1.2 of 2-D code released.
- 3-91 Preliminary Version 1.0 of 3-D code ready for testing.

PROTEUS NAVIER-STOKES CODE 2D Version



PROTEUS NAVIER-STOKES CODE 3D Version



PROTEUS NAVIER-STOKES CODE

CURRENT STATUS

- Version 1.2 of 2-D code available.
- 3-D code and documentation being upgraded.
- Two-equation k - ϵ turbulence model available in 2-D code.
- Grid patching (embedded boundary) capability available for 2-D code.
- Validation studies underway.
 - Laminar/turbulent compressible flat plate boundary layers
 - Laminar/turbulent flow past an airfoil.
 - Turbulent flow in transonic diffusers.

PROTEUS NAVIER-STOKES CODE

NEAR-TERM PLANS

- Complete upgrade of 3-D code and documentation.
- Investigate additional algebraic and pde turbulence models.
- Create diagonalized version.
- Continue validation of 2-D code, emphasizing turbulent flow and flow with heat transfer.
- Begin more intensive validation of 3-D code.
- Release 3-D code in fall of 1991.

PROTEUS NAVIER-STOKES CODE

LONG-TERM PLANS

- Continue validation of 2-D and 3-D codes.
- Add zonal gridding capability.
- Investigate convergence acceleration techniques.
- Add algebraic Reynolds stress turbulence model.
- Investigate adaptive mesh schemes.
- Hold user's workshop after 3-D code release.

ACCURATE UPWIND-MONOTONE (NONOSCILLATORY) METHODS FOR CONSERVATION LAWS

Hung T. Huynh
Internal Fluid Mechanics Division
NASA Lewis Research Center

The well-known MUSCL scheme of Van Leer (J. Comp. Phys., 1979)—a second-order extension of Godunov's first-order upwind-difference method—is constructed using a piecewise linear approximation. In order that the solutions are free from oscillations, a constraint is imposed on the slopes of the piecewise linear reconstruction. The MUSCL scheme is second-order accurate at the smooth part of the solution except at extrema where the accuracy degenerates to first-order due to the monotonicity constraint. Harten and Osher (SIAM J. Numer. Anal., 1987) resolved this loss of accuracy by introducing the concept of nonoscillatory reconstruction. Their UNO2 scheme involves no constraint, is uniformly second-order accurate, and does not create oscillations. The UNO2 scheme is, however, very diffusive, e.g., it smears contact discontinuities.

To construct accurate schemes which are free from oscillations, we first introduce the concept of upwind-monotonicity. Next, in a geometric framework in which the median function plays a crucial role, we present Van Leer's constraint and the UNO2 scheme. In this framework, the constraint is extended in such a manner that extrema are not "clipped", while upwind-monotonicity is preserved. At monotone part of the data, the new constraints reduce to that of Van Leer for the second-order case, and in the third order case, that of Colella and Woodward (J. Comp. Phys., 1984). Several classes of schemes, which are upwind-monotone and of uniform second or third-order accuracy, are then presented. These schemes also satisfy the nonoscillatory property of the UNO2 scheme. It is shown that the MUSCL, UNO2 and all the popular TVD (total variation diminishing) schemes (Harten, J. Comp. Phys., 1983; or Sweby, SIAM J. Numer. Anal., 1984) are members of the new class. Moreover, each TVD scheme corresponds to a uniform second-order accurate upwind-monotone scheme. The gain of accuracy is obtained by a few extra lines of Fortran programming. These new schemes are also extensions of the authors SONIC schemes (NASA TM 102010, 1989; also Yokota & Huynh, NASA TM 102354, 1990)

Results for advection with constant speed are shown below. The initial profile consists of a \sin^2 wave, a square wave, a triangle wave and a semi-ellipse wave. Each wave contains 20 grid points on a uniform grid of 200 points. Using periodic boundary conditions, the profile is advected one period with CFL number 0.5. The results after 400 time steps are shown in Figs. (A) for a member of the new class, (B) for the UNO2, (C) for the PPM (Colella and Woodward, J. Comp. Phys., 1984), and (D) for Roe's "Superbee" schemes. It can be seen that the new scheme compares favorably with the state-of-the-art methods.

Upwind Shock-Capturing Methods

Why new methods?

TVD (Total Variation Diminishing) schemes

- + Capture shocks with high resolution
- First-order accurate near extrema

UNO2 scheme

- + Uniformly second-order accurate
- + Nonoscillatory, i.e., no new strict extrema
- Somewhat diffusive

New Class of Methods

New geometric framework

Existing schemes (TVD, UNO)

are presented in the new framework

New schemes

- * Uniformly second-order accurate
- * Upwind-monotone and nonoscillatory
- * Extensions of TVD and UNO2 schemes

$$\frac{\partial u}{\partial \tau} + a \frac{\partial u}{\partial x} = 0, \quad a > 0$$

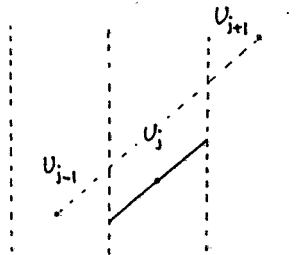
$$u(x, 0) = u_0(x).$$

Approach

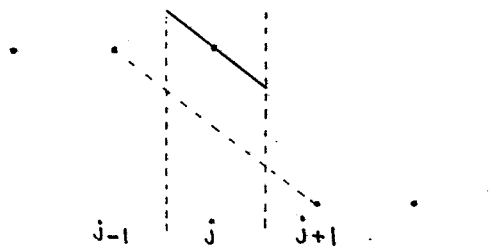
Projection or Reconstruction,

Evolution or Upwinding + Time stepping

The task for piecewise linear reconstruction:
 given the data $\{u_j\}$, define $(\text{slope})_j$

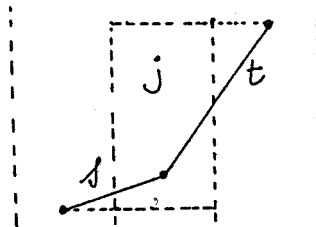


The problem: discontinuity



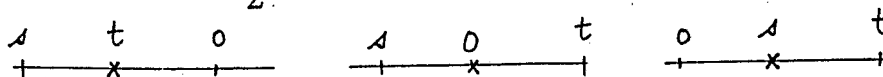
Van Leer's constraint

The linear reconstruction in cell j lies in
 $[\min(u_{j-1}, u_j, u_{j+1}), \max(u_{j-1}, u_j, u_{j+1})]$

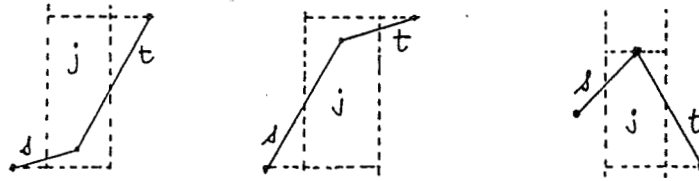


$$m(s, t) = \text{median}(s, t, 0)$$

$$= \frac{1}{2} [\text{sgn}(s) + \text{sgn}(t)] \min(|s|, |t|)$$



The pictures



Constraint: $(\text{slope})_j$ lies between 0 and $2r$,

$$r = m(s, t).$$

Algorithm: $(\text{slope})_j \leftarrow m[(\text{slope})_j, 2r]$



TVD schemes

Minmod scheme

$$(\text{slope})_j = r = m(s, t)$$

Average limiter

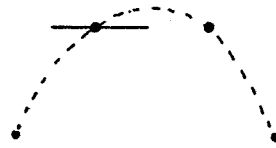
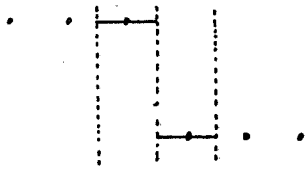
$$(\text{slope})_j = m\left(\frac{s+t}{2}, 2r\right)$$

Superbee scheme

$$(\text{slope})_j = m[\text{maxmod}(s, t), 2r]$$

$$\text{maxmod}(s, t) = \frac{1}{2}[\text{sgn}(s) + \text{sgn}(t)] \max(|s|, |t|)$$

The Peter-Paul Principle



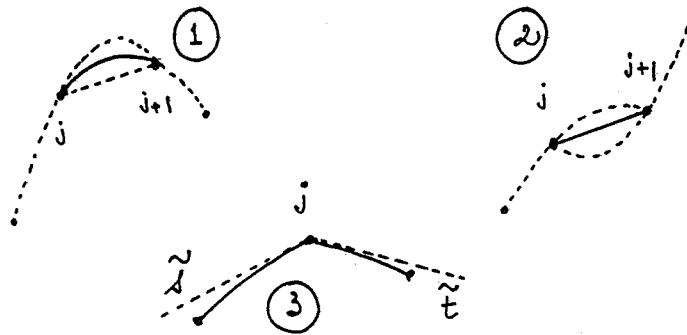
Resolve shock well (Paul), 1st-order at extrema (Peter).

For TVD schemes, u_j^{n+1} lies between u_j^n and u_{j-1}^n .

The data are **monotone** in $[x_j, x_{j+1}]$ if
 $u_{j-1} \leq u_j \leq u_{j+1} \leq u_{j+2}$, or $u_{j-1} \geq u_j \geq u_{j+1} \geq u_{j+2}$.

A scheme is **upwind-monotone** if monotonicity of the data in $[x_{j-1}, x_j]$ implies u_j^{n+1} lies between u_j and u_{j-1} .

UNO2 scheme



$$(\text{slope})_j = \tilde{r} = m(\tilde{s}, \tilde{t})$$

- + Second-order accurate
- + Nonoscillatory
- Diffusive

New methods

New constraint:

$$(\text{slope})_j \text{ lies in } I[0, 2r, \tilde{r}] = I[0, r_{\max}]$$

$$r_{\max} = \text{sgn}(\tilde{r}) \max(2|r|, |\tilde{r}|).$$

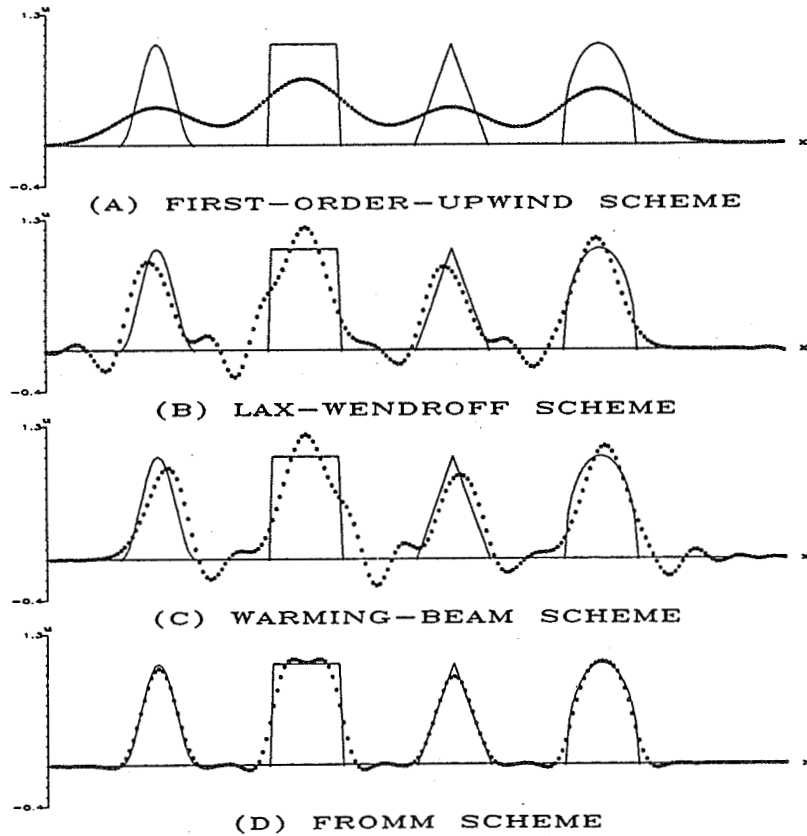
New algorithm: $(\text{slope})_j \leftarrow m[(\text{slope})_j, r_{\max}]$.

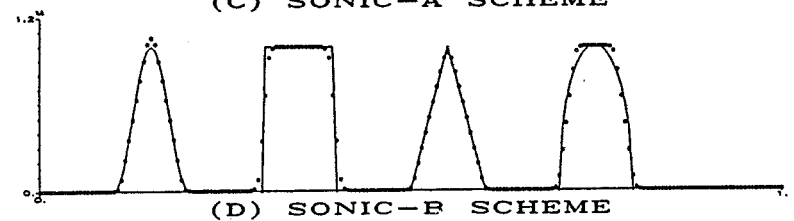
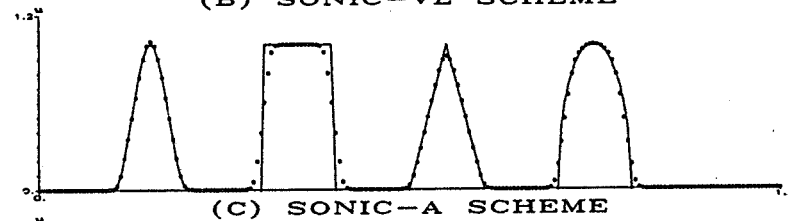
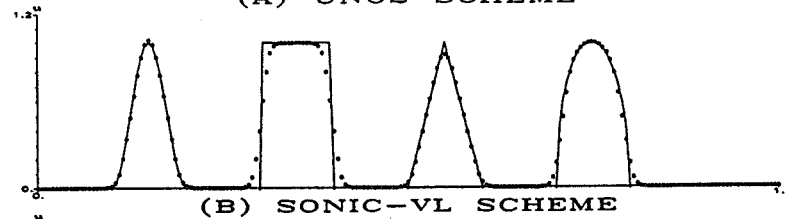
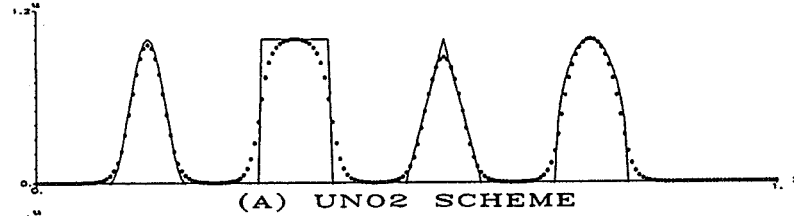
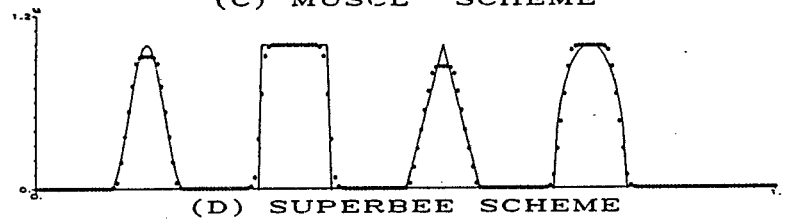
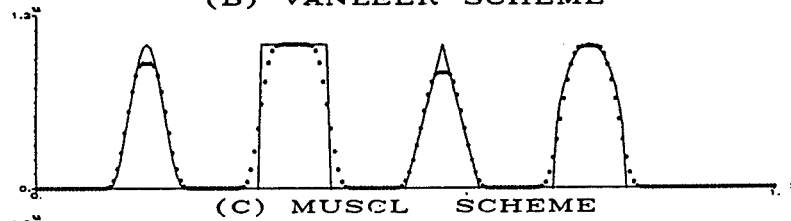
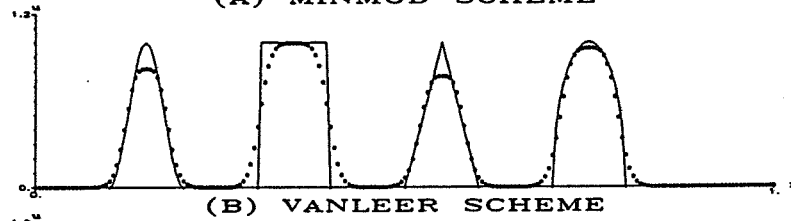
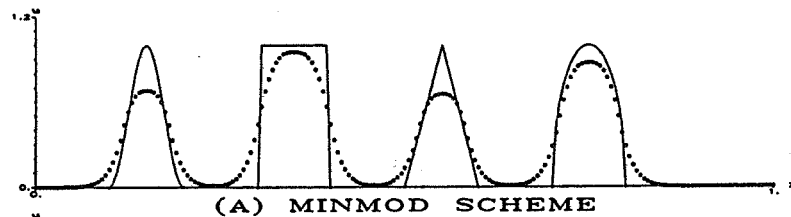
$$\text{UNO } (\text{slope})_j = \tilde{r} = m(\tilde{s}, \tilde{t})$$

$$\text{SONIC-A } (\text{slope})_j = m\left(\frac{\tilde{s} + \tilde{t}}{2}, r_{\max}\right)$$

$$\text{SONIC-B } (\text{slope})_j = m[\text{maxmod}(\tilde{s}, \tilde{t}), r_{\max}]$$

Second-order accurate and upwind-monotone





Conclusions

A new framework for shock-capturing methods was presented

- . Simple
- . Geometric
- . Uses the median function

The approach leads to new classes of methods:

- . Compare favorably with existing methods
- . Resolve discontinuities with high resolution
- . Uniformly second-order accurate
- . Upwind-monotone (therefore, nonoscillatory)
- . Generalization of TVD and UNO2 schemes
- . Vectorize on Cray

Numerical Simulation of Conservation Laws

Sin Chung-Chang
Internal Fluid Mechanics Division
NASA Lewis Research Center

Wai Ming To
Sverdrup Technology, Inc.
Lewis Research Center Group
NASA Lewis Research Center

A new numerical framework for solving conservation laws is being developed. This new approach differs substantially from the well established methods, i.e., finite difference, finite volume, finite element and spectral methods, in both concept and methodology. The key features of the current explicit scheme include: (a) direct discretization of the integral forms of conservation laws, (b) treating space and time on the same footing, (c) flux conservation in space and time, and (d) unified treatment of the convection and diffusion fluxes.

The model equation considered in the preliminary study is the standard 1-D unsteady constant-coefficient convection-diffusion equation. In a stability study, it is shown that the principal and spurious amplification factors of the current scheme, respectively, are structurally similar to those of the leapfrog/DuFort-Frankel scheme. As a result, the current scheme has no numerical diffusion in the special case of pure convection and is unconditionally stable in the special case of pure diffusion.

Assuming smooth initial data, it will be shown theoretically and numerically that, by using an easily determined optimal time-step, the accuracy of the current scheme may reach a level which is several orders of magnitude higher than that of the MacCormack scheme, with virtually identical operation count.

Introduction

The focus of the current method development is entirely on the integral form of conservation laws. Key features of the current **explicit** scheme include:

- Direct discretization of the integral forms of conservation laws.
- Treating space and time on the same footing.
- Flux conservation in space and time.
- Unified treatment of the convection and diffusion fluxes.
- Low-operation count and high accuracy.

Unsteady 1-D Convection-Diffusion Equation

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} - \mu \frac{\partial^2 u}{\partial x^2} = 0$$

or

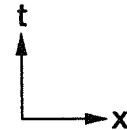
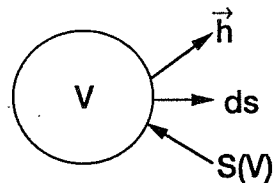
$$\vec{\nabla} \cdot \vec{h} = 0$$

with

$$\vec{h} \stackrel{\text{def}}{=} \left(au - \mu \frac{\partial u}{\partial x}, u \right)$$

or

$$\oint_{S(V)} \vec{h} \cdot d\vec{s} = 0$$

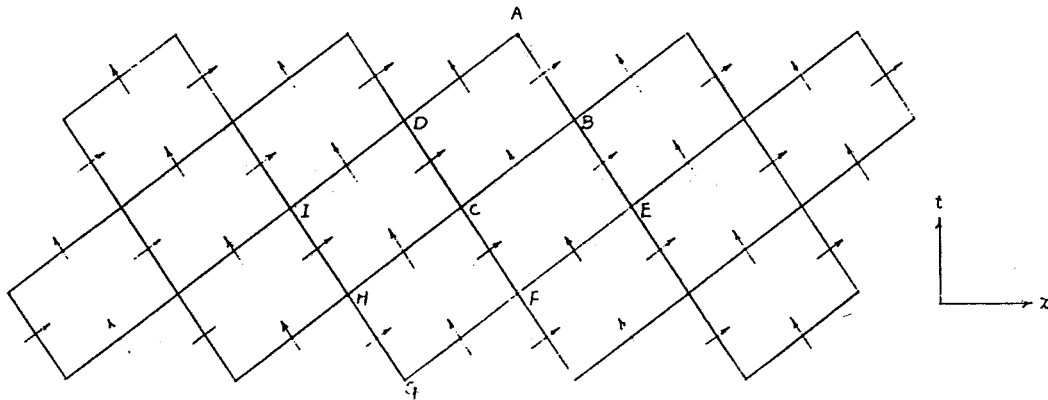


V = Any volume in space-time.

$S(V)$ = Boundary of V .

$\vec{h} \cdot d\vec{s}$ = Flux of \vec{h} leaving V through surface element $d\vec{s}$.

Basic Conservation Domains (BCD's)



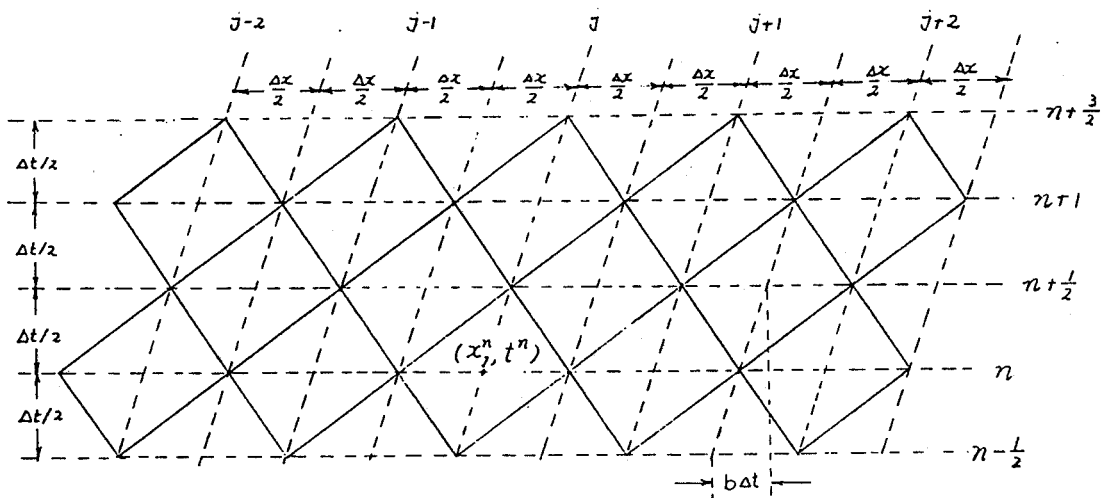
For each BCD, it is required that

$$\text{Sum of incoming fluxes} = \text{sum of outgoing fluxes}$$

<1>

As a result, <1> is also true for the union of any collection of BCD's.

Space - Time Mesh and BCD's



$$b \stackrel{\text{def}}{=} \left(\frac{dx}{dt} \right) \text{ along a } j \text{ mesh line}$$

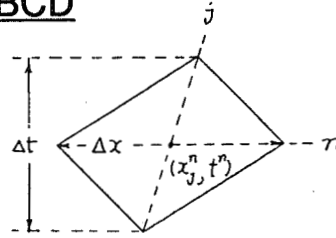
$$x_j^n = j\Delta x + n \cdot b\Delta t, \quad t^n = n\Delta t$$



The Solution in a BCD

$D(j,n) = \text{BCD centered at } (x_j^n, t^n).$

$D'(j,n) = \text{Interior of } D(j,n).$



$$\begin{aligned} \underline{u}(x,t; j,n) &= \text{Approximation of } u(x,t) \text{ in } D'(j,n) \\ &= \alpha_j^n (x - x_j^n) + \beta_j^n (t - t^n) + \gamma_j^n \end{aligned}$$

Since total flux leaving $D'(j,n) = 0$

$$\beta_j^n = -a \alpha_j^n$$

Thus

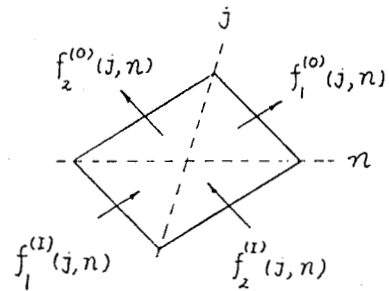
$$\underline{u}(x,t; j,n) = \alpha_j^n [(x - x_j^n) - a(t - t^n)] + \gamma_j^n$$

Fundamental Relations (1)

$$\underline{u}(x,t; j,n) = \alpha_j^n [(x - x_j^n) - a(t - t^n)] + \gamma_j^n$$

$$\underline{u}(x_j^n, t^n; j,n) = \gamma_j^n$$

$$\frac{[\partial \underline{u}(x,t; j,n)]}{\partial x} \bigg|_{x=x_j^n, t=t^n} = \alpha_j^n$$



$$\tau \stackrel{\text{def}}{=} \frac{(a-b)\Delta t}{\Delta x}$$

$$\delta \stackrel{\text{def}}{=} \frac{4\mu\Delta t}{(\Delta x)^2}$$

(τ, δ)

$$\frac{\Delta x}{2} \gamma_j^n, \frac{(\Delta x)^2}{8} \alpha_j^n$$

(τ, δ)

$$f_1^{(1)}(j,n), f_2^{(1)}(j,n)$$

(τ, δ)

$$f_1^{(0)}(j,n), f_2^{(0)}(j,n)$$

Invariant Property (1)

Let

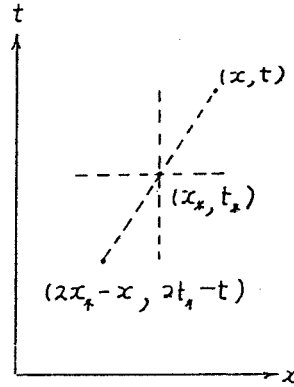
$$u = u_0(x, t; a, \mu) \quad <1>$$

be a solution of

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} - \mu \frac{\partial^2 u}{\partial x^2} = 0 \quad <2>$$

Let x_* and t_* be any real constants. Then

$$u = u_0(2x_* - x, 2t_* - t; a, -\mu) \quad <3>$$



is also a solution of <2>. As an example, consider

$$u_0(x, t; a, \mu) \stackrel{\text{def}}{=} e^{-4\pi^2 \mu t} \sin [2\pi(x - at)] \quad <4>$$

The above property may be referred to as the p-t- μ (parity-time reversal- μ reversal) invariance.

Invariant Property (2)

The main scheme of the Leapfrog/DuFort-Frankel method i.e.,

$$\frac{u_j^{n+1} - u_j^{n-1}}{2\Delta t} + a \frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} = \mu \frac{u_{j+1}^n - u_j^{n+1} - u_j^{n-1} + u_{j-1}^n}{(\Delta x)^2} \quad <5>$$

is also p-t- μ invariant. In other words, if

$$u_j^n = u_0(j, n; a, \mu)$$

satisfy <5>, then so does (j_* and n_* are constant integers)

$$u_j^n = u_0(2j_* - j, 2n_* - n; a, -\mu)$$

The present scheme is p-t- μ invariant also. Other classical schemes generally are not p-t- μ invariant. p-t- μ invariance has an impact on the stability property.

Stability Condition

1. The stability condition for the present scheme is

$$1 \geq \tau^2$$

i.e., stability is not dependent on the viscosity μ .

2. The present scheme has no numerical diffusion in the special case of pure convection, and is unconditionally stable in the special case of pure diffusion.

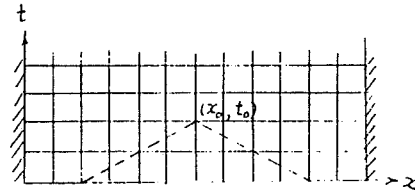
Inconsistency

- A smooth solution which satisfies the present or the Leapfrog/Dufort-Frankel scheme at the mesh points will satisfy

$$\partial u / \partial t + a \partial u / \partial x - \mu \partial^2 u / \partial x^2 = 0 \quad (\mu \neq 0) \quad <1>$$

as $\Delta x \rightarrow 0$ and $\Delta t / \Delta x \rightarrow 0$.

- A characteristic line of <1> is
 $t = \text{constant}$

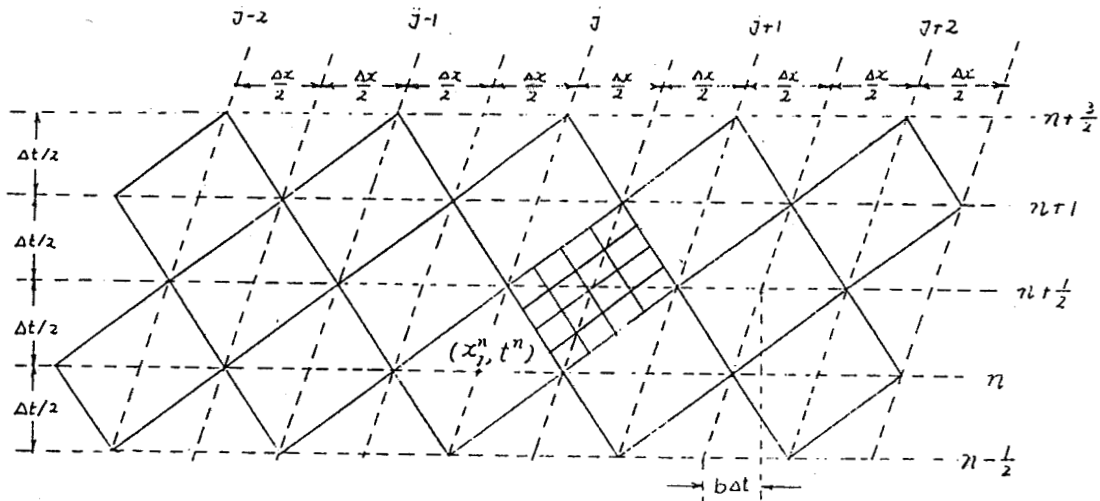


Thus a discrete solution may converge to an analytical solution only if $(\Delta t / \Delta x) \rightarrow 0$.

- For a typical classical scheme, stability requires that

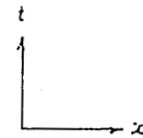
$$\frac{\mu \Delta t}{(\Delta x)^2} \equiv \frac{\mu (\Delta t / \Delta x)}{\Delta x} \text{ be bounded. Thus, } (\Delta t / \Delta x) \rightarrow 0 \text{ as } \Delta x \rightarrow 0$$

Space - Time Mesh and BCD's

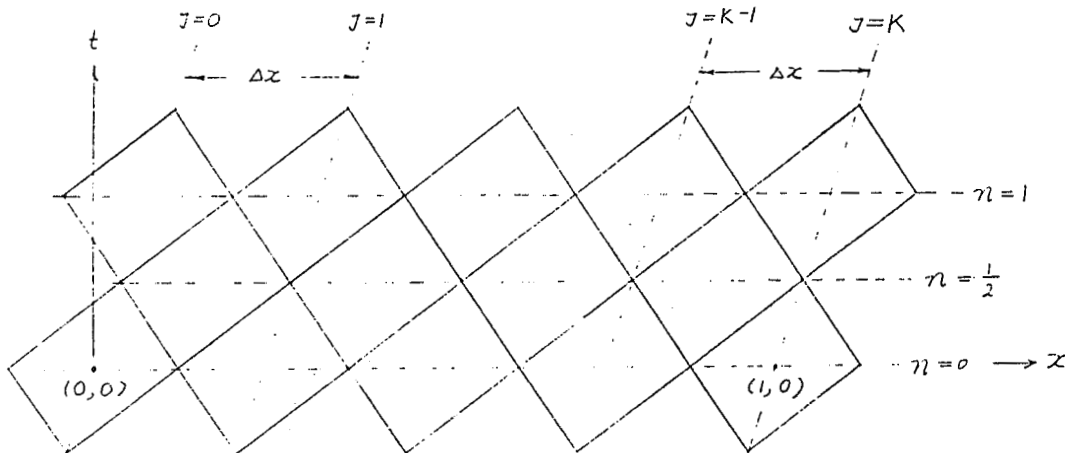


$b \stackrel{\text{def}}{=} \left(\frac{dx}{dt} \right)$ along a j mesh line

$$x_j^n = j\Delta x + n \cdot b\Delta t, \quad t^n = n\Delta t$$



Problem Definition (1)



Analytical Solution:

$$u = u_a(x, t) = e^{-4\pi^2 \mu t} \sin[2\pi(x - at)]$$

$$\gamma_j^0 = u_a(j\Delta x, 0) = \sin(2\pi j\Delta x), \quad \alpha_j^0 = \frac{\partial u_a(j\Delta x, 0)}{\partial x} = 2\pi \cos(2\pi j\Delta x)$$

$$\text{wavelength} = K\Delta x = 1 \quad u(x, t; j, n) = \alpha_j^n \left[(x - x_j^n) - a(t - t^n) \right] + \gamma_j^n$$

Problem Definition (2)

A problem is defined by specifying the values of (i) physical parameters a and μ , (ii) the mesh parameters b , Δt and $\Delta x (= 1/K)$, and (iii) total running time t . The total time-step number

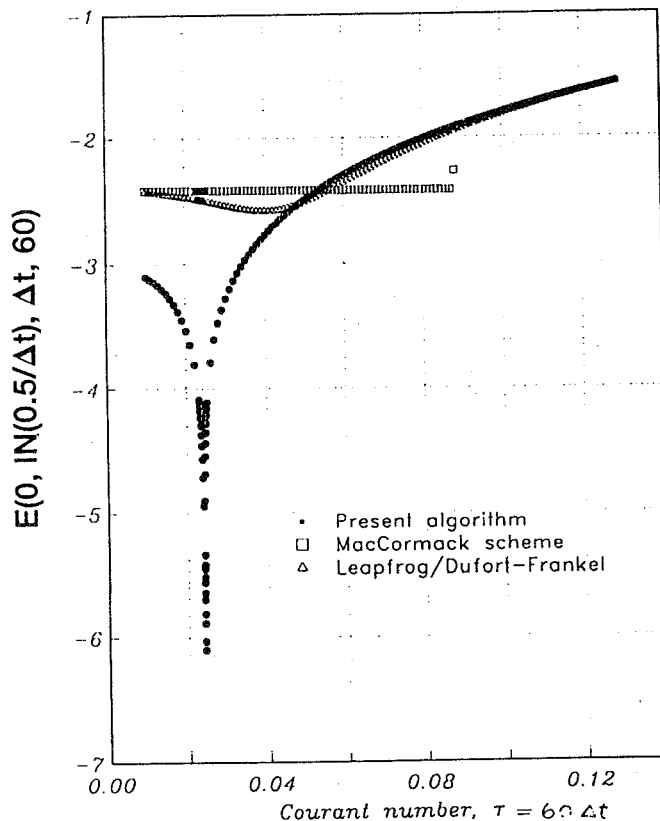
$n = \text{IN}(t/\Delta t) \stackrel{\text{def}}{=} \text{the integer nearest to the ratio } (t/\Delta t)$

Error of a problem is measured by:

$$E(b, n, \Delta t, K) \stackrel{\text{def}}{=} \log_{10} \left[\frac{1}{e^{-4\pi^2 \mu t}} \cdot \frac{1}{K} \sum_{j=1}^K |u_j^n - u_a(x_j^n, t^n)| \right]$$

Accuracy of present method is optimized when either

$$\boxed{\text{(i) } \tau = 0 \text{ or (ii) } 1 - \tau^2 = \sqrt{3} \delta. \quad \tau \stackrel{\text{def}}{=} (a-b)\Delta t / \Delta x, \delta \stackrel{\text{def}}{=} 4\mu \Delta t / (\Delta x)^2}$$



$$\begin{aligned} a &= 1, \quad \mu = 0.1 \\ b &= 0, \quad K = 60 \\ t &= 0.5 \\ \Delta t &\text{ varies} \end{aligned}$$

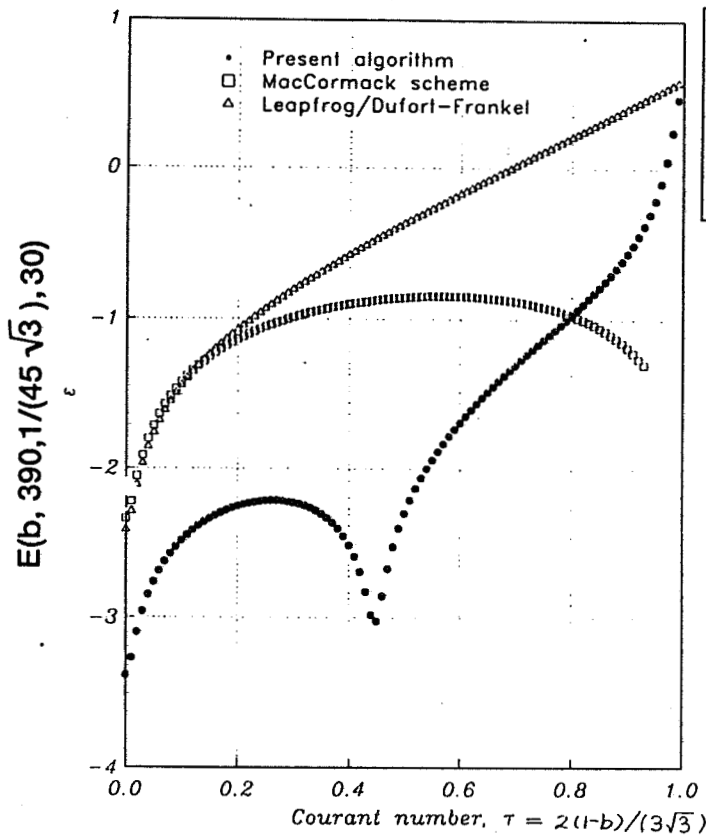
$$\text{Re} \stackrel{\text{def}}{=} \tau / \gamma$$

$$= \frac{(a-b)\Delta x}{4\mu} = \frac{1}{24}$$

$$\tau = \frac{(a-b)\Delta t}{\Delta x} = 60\Delta t$$

Optimization:

$$\boxed{1 - \tau^2 = \sqrt{3} \gamma}$$



$a = 1, \quad m = 0.01$
 $K = 30, \quad \Delta t = 1/(45\sqrt{3})$
 $t = 5$
 b varies

$$\delta = \frac{4\mu\Delta t}{(\Delta x)^2} = \frac{0.8}{\sqrt{3}}$$

$$\tau = \frac{2(1-b)}{3\sqrt{3}}$$

Optimization:

$$t = 0$$

or

$$1 - \tau^2 = \sqrt{3} \quad \gamma$$

Special Property

Let K be the number of spatial mesh points. Then the present marching scheme may be expressed as

$$\vec{u}^{n+1} = A \vec{u}^n + \vec{b}^n, \quad n = 0, 1, 2, \dots$$

Here (i) \vec{u}^n and \vec{u}^{n+1} , respectively, are the $2K \times 1$ column matrices representing $2K$ numerical variables at the time levels n and $n+1$, (ii) \vec{b}^n is a given $2K \times 1$ column matrix at the time level n , and (iii) A is a given $2K \times 2K$ sparse matrix.

Unusual property: A^{-1} is also sparse and is known explicitly

Conclusions (1)

- Flux conservation is the focus of method development.
- Unified treatment of space and time, also of convection and diffusion fluxes.
- Stability is not dependent on the viscosity μ . The present scheme has no numerical diffusion if $\mu = 0$ and is unconditionally stable if $a = 0$.
- The sparse matrix A in the present marching scheme $\vec{u}^{n+1} = A \vec{u}^n + \vec{b}^n$ has the unusual property that A^{-1} is also sparse and is known explicitly.
- The present scheme has the same invariant property of the physical equation it models.

Conclusions (2)

- The present scheme is most accurate and stable when $\tau = 0$, i.e., $b = a$
- For the present scheme, mesh refinement may be made locally without impacting global structure.
- Since initial data required include both u_j^0 and $(\partial u / \partial x)_j^0$, initial-value specification of the present method is more accurate than that of traditional methods.
- Assuming smooth, initial data, it is shown that, by using an easily determined optimal time-step, the accuracy of the current scheme may reach a level which is several orders of magnitude higher than the MacCormack scheme, with virtually identical operation count.

A New Lagrangian Method for Real Gases at Supersonic Speed

C.Y. Loh and Meng-Sing Liou
Internal Fluid Mechanics Division
NASA Lewis Research Center

Abstract

With the renewed interest in high-speed flights, the real gas effect is of theoretical as well as practical importance. In general, this real gas phenomenon is seen in the regions of high gradients and discontinuities. The ability of numerically capturing them *accurately*, viz with minimal numerical dissipation and oscillations, has been the main challenge to the computational fluid dynamist as well as numerical analyst over the past four decades. In the past decade, upwind splittings or Godunov-type Riemann solutions have received tremendous attention and as a result significant progress has been made both in the ideal and non-ideal gas. However, almost all of these efforts have been exclusively formulated in the framework of Eulerian description of fluid motion. Although the Lagrangian description has been used in several reports, they are always followed by a remap step from a Lagrangian grid back to the original Eulerian one. This not only involves tedious interpolation procedure but also can corrupt the accuracy which was gained by the choice of the Lagrangian approach over the Eulerian one.

However, since all upwind schemes used by today's major CFD codes are strictly formulated in 1-D framework, and only formally extended to multi-dimensions in space. Consequently, the attractive property of crisp representation of discontinuities is lost and search for genuine multi-dimensional approach has just been undertaken by several leading researchers, but still based on Eulerian description. In this paper, we propose a new alternative that is formulated using the Lagrangian description, for the calculation of supersonic/hypersonic real-gas inviscid flows. This new formulation avoids grid generation step which is automatically obtained as solution procedure marches in the "time-like" direction. As a result, no remapping is required and the accuracy is faithfully maintained in the Lagrangian level. Moreover, the ability of *preserving* the sharpness of discontinuities in multi-dimensions is vividly demonstrated. In this paper, we will give numerical results for a variety of real-gas problems consisting of essential elements in the high-speed flows, such as shock waves, expansion waves, slip surfaces and their interactions. Finally, calculations for flows in a generic inlet and a nozzle are presented.

Outline

- . New Lagrangian formulation and the conservation form for 2-D steady supersonic flow (for perfect gas)
- . Advantages of the new Lagrangian method
- . Real gas version of the new Lagrangian method
- . Godunov scheme and TVD scheme
- . Numerical examples for supersonic real gas flow
- . Concluding Remark

New Lagrangian formulation and the conservation form for 2-D steady supersonic flow (perfect gas)

Independent variables of Eulerian formulation: x, y

Independent variables of the new Lagrangian formulation: τ, ξ
(Hui and Van Roessel, 1985)

τ — — — *Lagrangian time*

ξ — — — *stream function*

Lagrangian conservation form: (Loh and Hui, J.C.P., V.89, 207, 1990)

$$\frac{\partial \mathbf{E}}{\partial \tau} + \frac{\partial \mathbf{F}}{\partial \xi} = 0 \quad (1)$$

$$\mathbf{E} = \begin{pmatrix} K \\ H \\ Ku + pV \\ Kv - pU \\ U \\ V \end{pmatrix}, \quad \mathbf{F} = \begin{pmatrix} 0 \\ 0 \\ -pv \\ pu \\ -u \\ -v \end{pmatrix}$$

where

$$u = \frac{\partial x}{\partial \tau}, \quad v = \frac{\partial y}{\partial \tau}$$

$$U = \frac{\partial x}{\partial \xi}, \quad V = \frac{\partial y}{\partial \xi}$$

$$K = \rho(uV - vU)$$

$$H = (u^2 + v^2)/2 + h(p, \rho)$$

$$h(p, \rho) = \frac{\gamma p}{(\gamma - 1)\rho}$$

Advantages of the new Lagrangian method

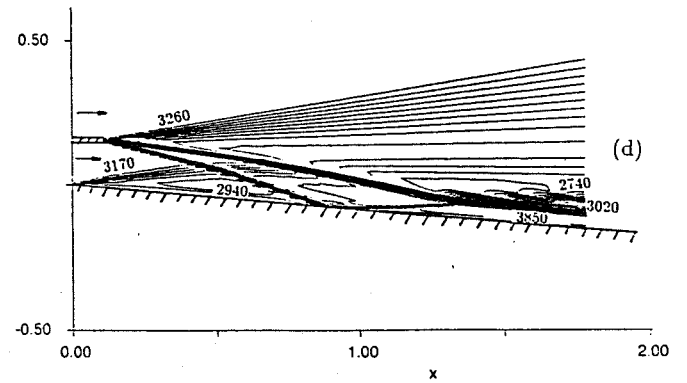
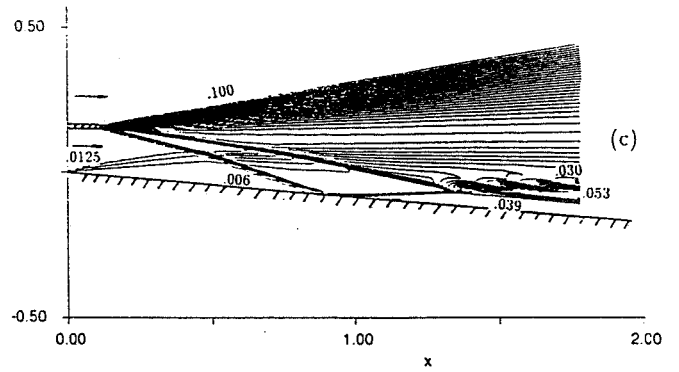
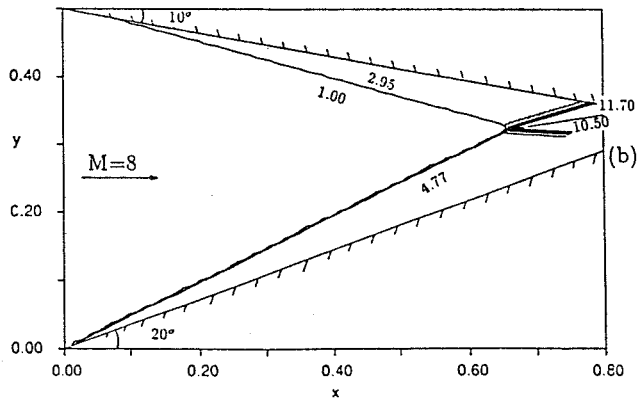
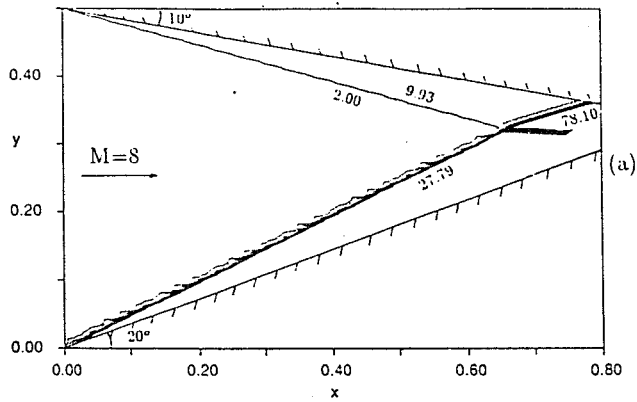
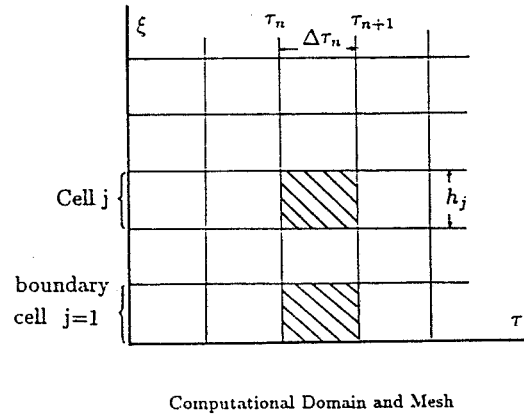
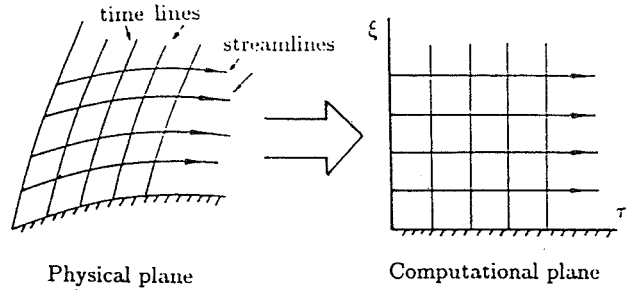
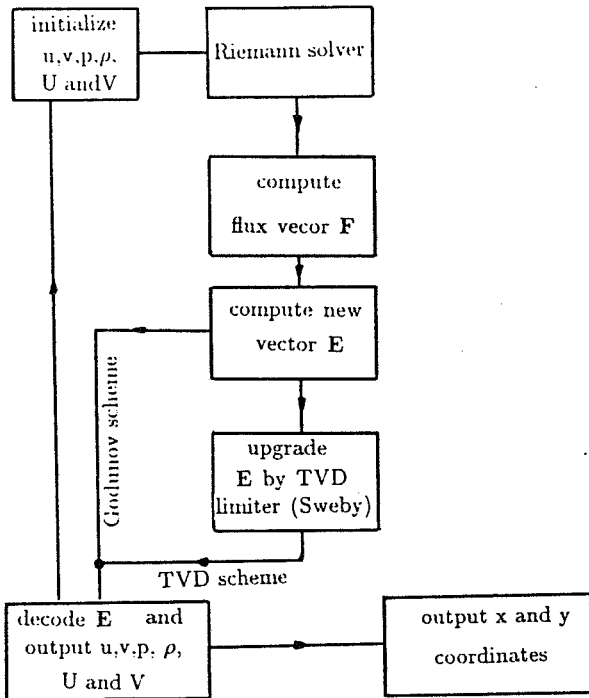
- (1) Truly multi-dimensional computation (no time splitting), a computational cell is literally a fluid particle, physics is closely followed.
- (2) Crisp slip line (contact) resolution.(1 point)
- (3) No grid generation is needed, grids are automatically generated following streamlines.
- (4) Particularly appropriate for hypersonic computation
- (5) Ready for parallel computation.

Real gas version of the new Lagrangian method

- . Accurate real gas Riemann solver
- . Decoding of \mathbf{E} to find u, v, p, ρ

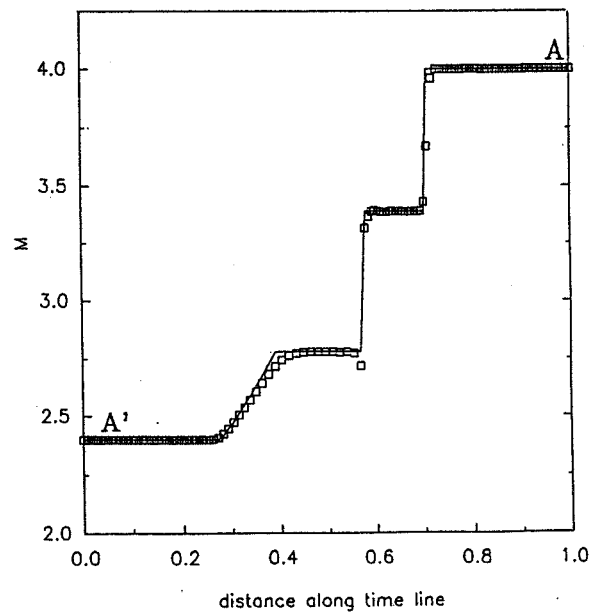
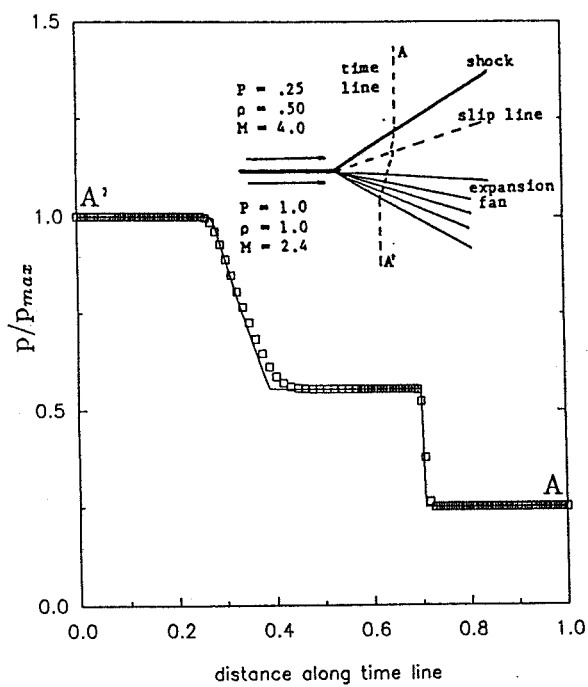
Tannehill's EOS is used (NASA CR 2470, 1974)

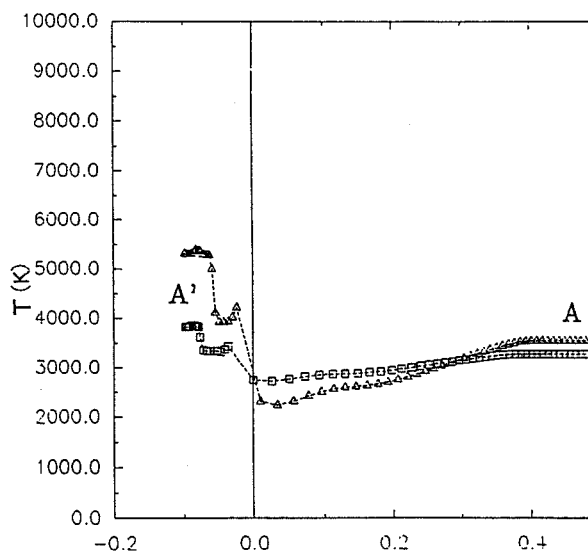
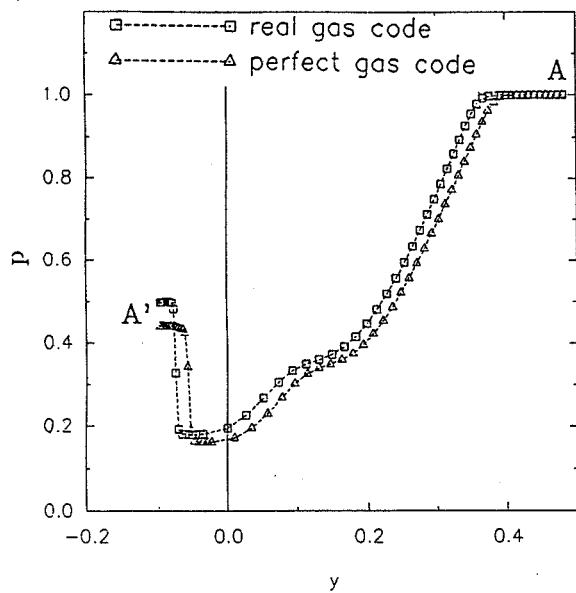
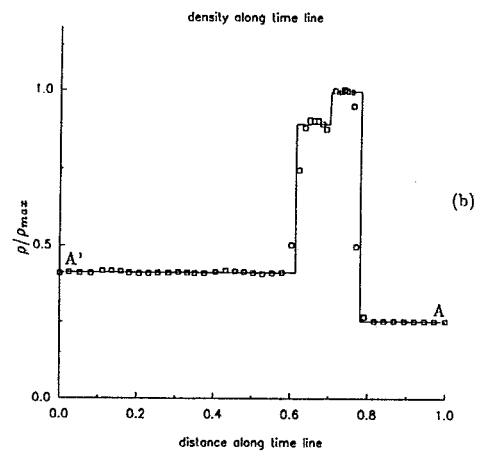
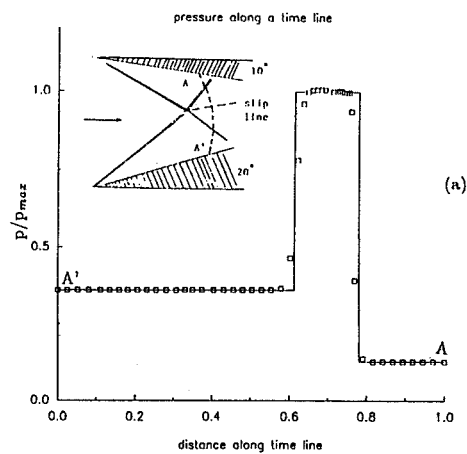
Godunov/TVD scheme

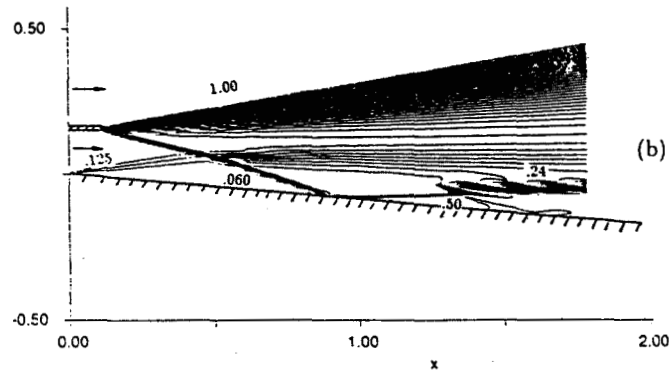
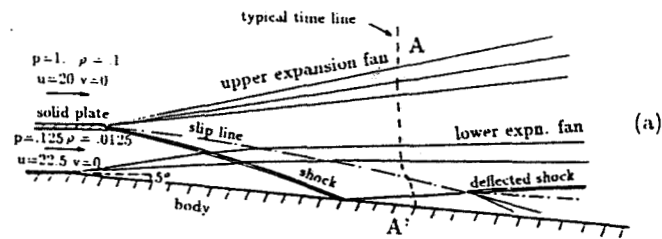


Numerical examples for real gas supersonic flows

- . Riemann problem
- . Shock collision in converging channel
- . Generic nozzle flow







Pressure, density and temperature contours for a sophisticated real gas outlet problem, showing interactions among the waves and shock reflection on the body surface; (a) schematic sketch, (b) isobars, (c) isopycnics and (d) isotherms.

Concluding Remarks

Accurate yet efficient real gas effect computation:

- . As accurate as Tannehill's EOS warrants
- . Genuine steady approach saves CPU
- . Bears all the advantages of the Lagrangian method mentioned above

A TIME-ACCURATE IMPLICIT METHOD FOR CHEMICAL
NON-EQUILIBRIUM FLOWS AT ALL SPEEDS

Jian-Shun Shuen
Sverdrup Technology, Inc.
Lewis Research Center Group
NASA Lewis Research Center

Summary

A new time-accurate coupled solution procedure for solving the chemical non-equilibrium Navier-Stokes equations over a wide range of Mach numbers is described. The scheme is shown to be very efficient and robust for flows with velocities range from $M \leq 10^{-10}$ to supersonic speeds.

Description

Chemically reacting flows in aeropropulsion systems are often not amenable to numerical algorithms developed for high speed compressible flows. Examples include the rocket motor flow which involves a wide range of Mach numbers (from near zero velocity at the closed end to supersonic at the nozzle exit) and flow in gas turbine combustor where velocity is in the low subsonic range, yet the variation in density is very large so as to preclude an incompressible approach. Unfortunately, most available solution methods are usually applicable to either compressible flow with moderate to high Mach numbers or incompressible flows, but not both. This difficulty points out the need for a single solution technique that can be employed in flows with wide Mach number range and large property variations.

There are two underlying reasons for the difficulty of low Mach number computations in compressible flows. When the Mach number becomes low, the eigenvalues of the system differ widely so that the equations becomes stiff, resulting in significant slowdown in the convergence rate. Another reason is the singular behavior of the pressure term in the momentum equations as Mach number approaches zero, causing large roundoff error and smearing the pressure variation field, and, consequently, preventing the convergence.

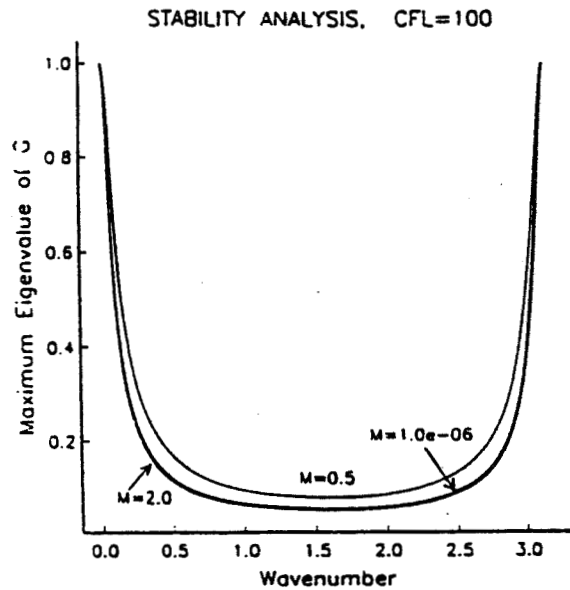
In the present numerical algorithm, methods to overcome these difficulties are proposed and successfully tested. The resulting code is then extended to consider the effects of non-equilibrium chemistry and real-gas properties. Included in the abstract are results from a linear stability analyses for the present algorithm (referred to as the pressure based scheme) and a conventional compressible flow algorithm (density-based algorithms), and convergence histories for the convergent-divergent nozzle flows. The transonic nozzle flow has an inlet Mach number of 0.03 and exit Mach number 3.0. The subsonic cases are obtained by adjusting the nozzle back pressure so that flow is not choked at the throat. These results demonstrate the efficiency of the present method for flows over a wide Mach number range.

MOTIVATION

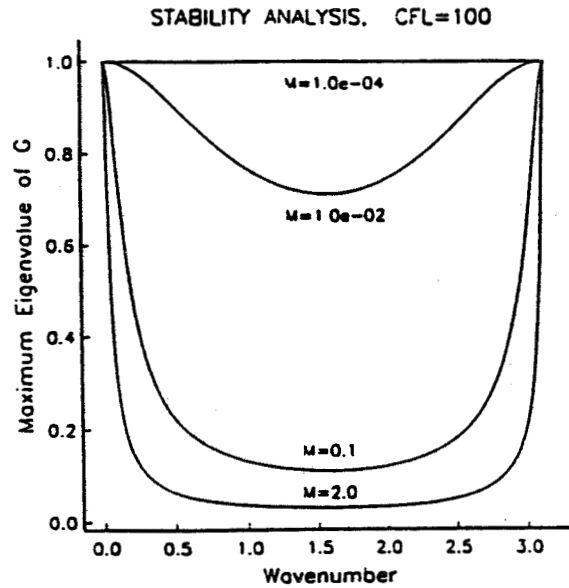
- Reacting flows in aeropropulsion devices are often not amenable to numerical algorithms developed for high speed compressible flows, e.g.,
 - rocket motor – wide range of Mach numbers, from near zero velocity at closed end to supersonic flow at nozzle exit.
 - gas turbine combustor – low subsonic velocities, but large variation in density precludes incompressible approach.
- Most popular codes for gas turbine combustor and other low-speed reacting flows are TEACH-type codes, which were based on technologies developed more than 20 years ago.
- Compressible flow CFD technologies can be carried over to low-speed flow regime.

OBJECTIVES

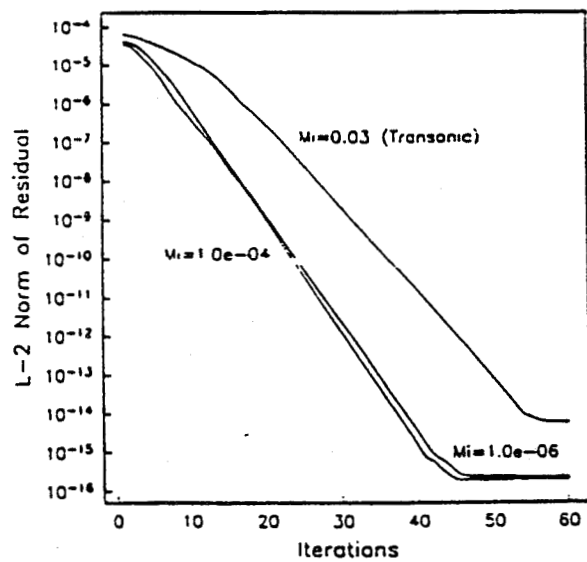
1. Development of an efficient and robust Navier-Stokes code for chemically reacting flows at **all speeds**.
2. Implementation of liquid-fuel spray combustion and turbulent combustion closure models.
3. Demonstration of the capability of the code for aeropropulsion applications, including flows in gas turbine combustor, after burner, liquid propellant rocket combustion chamber, etc.



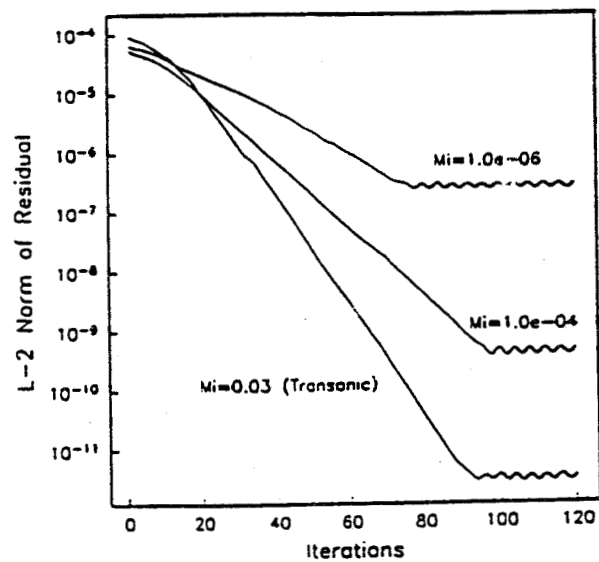
Magnitude of the Maximum Eigenvalue of the Amplification Matrix G , Pressure-Based Scheme, CFL=100.



Magnitude of the Maximum Eigenvalue of the Amplification Matrix G , Density-Based Scheme, CFL=100.



Convergence History of the Pressure-Based Scheme for the C-D Nozzle Flow, CFL=100.



Convergence History of the Density-Based Scheme for C-D Nozzle Flow, CFL=100.

Difficulties with Compressible Flow Algorithms at Low Mach Numbers

- Disparities among system's eigenvalues (stiffness), u , $u + c$, $u - c$, resulting in significant slowdown in convergence rate.
- Singular behavior of pressure gradient term in momentum equations as Mach number approaches zero,

$$\rho^* u^{*2} + \frac{p^*}{\gamma M_r^2}$$

As Mach number is decreased, pressure variation ($\Delta p^* \propto M^2$) becomes of similar magnitude as roundoff error of the large pressure gradient term ($p^*/\gamma M_r^2$).

METHOD OF APPROACH

Pressure Singularity Problem

- Pressure decomposed into two parts:

$$p = p_o + p_g$$

p_g replaces p in momentum equations and retains p_o as one of the unknowns.

- Employs conservative form of the governing equations, but uses primitive variables

$$(p_g, u, v, h, Y_i)$$

as unknowns. Conservation property preserved for flows at moderate to high Mach numbers. Pressure field accurately resolved for low Mach number flows.

Eigenvalue Stiffness Problem

- Pressure rescaled so that all eigenvalues have the same order of magnitude. Physical acoustic waves removed and replaced by a set of pseudo-acoustic waves which travel at speed comparable to fluid convective velocity.

Discretized Equations

$$\{\Gamma - \Delta\tau D + \frac{3}{2} \frac{\Delta\tau}{\Delta t} T + \Delta\tau (\frac{\partial A}{\partial x} - \frac{\partial^2 A_v}{\partial x^2})\} \Delta \tilde{Q} = -\Delta\tau (R)^p$$

where

$$R = \frac{3Q^p - 4Q^n + Q^{n-1}}{2\Delta t} + \frac{\partial(E - E_v)}{\partial x} - H$$

$$A = \frac{\partial E}{\partial \tilde{Q}}, \quad A_v = \frac{\partial E_v}{\partial \tilde{Q}},$$

$$D = \frac{\partial H}{\partial \tilde{Q}}, \quad T = \frac{\partial Q}{\partial \tilde{Q}}.$$

- Use dual-time stepping technique for time-accurate solutions.
- Fully implicit, fully coupled solution method. Use LU factorization for multi-dimensional flows.

ONE DIMENSIONAL GOVERNING EQUATIONS

$$\Gamma \frac{\partial \tilde{Q}}{\partial \tau} + \frac{\partial Q}{\partial t} + \frac{\partial(E - E_v)}{\partial x} = H$$

$$\tilde{Q} = A_f \begin{pmatrix} p_g \\ u \\ h \\ Y_1 \\ \vdots \\ Y_{N-1} \end{pmatrix} \quad Q = A_f \begin{pmatrix} \rho \\ \rho u \\ \rho e \\ \rho Y_1 \\ \vdots \\ \rho Y_{N-1} \end{pmatrix} \quad \Gamma = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & \dots & 0 \\ u & p & 0 & 0 & 0 & \dots & 0 \\ h & p & p & 0 & 0 & \dots & 0 \\ Y_1 & 0 & 0 & \rho & 0 & \dots & 0 \\ Y_2 & 0 & 0 & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & 0 \\ Y_{N-1} & 0 & 0 & 0 & \dots & 0 & \rho \end{pmatrix}$$

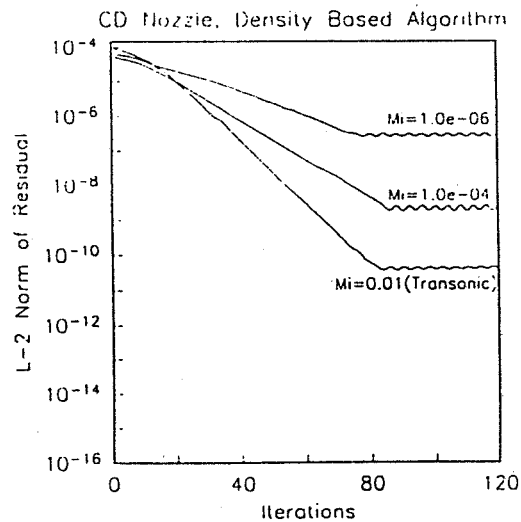
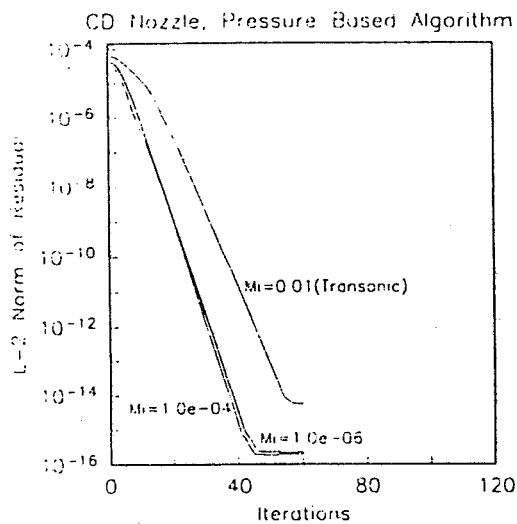
$$E = A_f \begin{pmatrix} \rho u \\ \rho u^2 + p_g \\ (\rho e + p)u \\ \rho u Y_1 \\ \vdots \\ \rho u Y_{N-1} \end{pmatrix} \quad E_v = A_f \begin{pmatrix} 0 \\ \frac{4}{3} \mu \frac{\partial u}{\partial x} \\ q_e \\ q_1 \\ \vdots \\ q_{N-1} \end{pmatrix} \quad H = \begin{pmatrix} 0 \\ p_g \frac{\partial A_f}{\partial x} \\ 0 \\ A_f \dot{\omega}_1 \\ \vdots \\ A_f \dot{\omega}_{N-1} \end{pmatrix}$$

$$\lambda_1 = u \quad \lambda_{2,3} = \frac{1}{2} [\alpha \pm \sqrt{\alpha^2 - 4\beta(M^2 - 1)}] \quad \alpha = u \left(\frac{\gamma\beta}{c^2} + 1 \right)$$

$$\beta = \frac{u^2}{4} \quad \text{As } M \longrightarrow 0, \quad \lambda_{max}/\lambda_{min} = 4.8$$

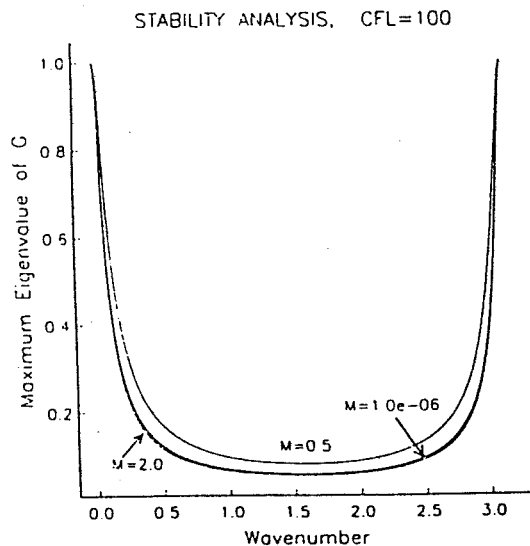
CONVERGENCE HISTORY FOR THE C-D NOZZLE FLOW, CFL=100.

Non-Equilibrium Air Dissociation/Recombination Chemistry
5 Species, 11 Reaction Steps

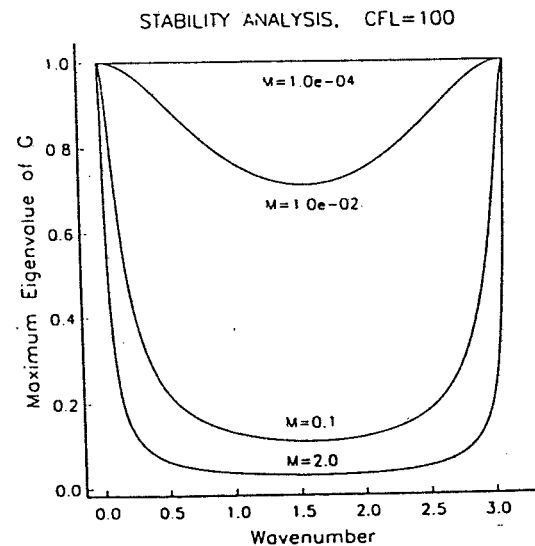


STABILITY ANALYSIS

Linear stability analysis of 1-D inviscid equations using Euler implicit time marching technique.



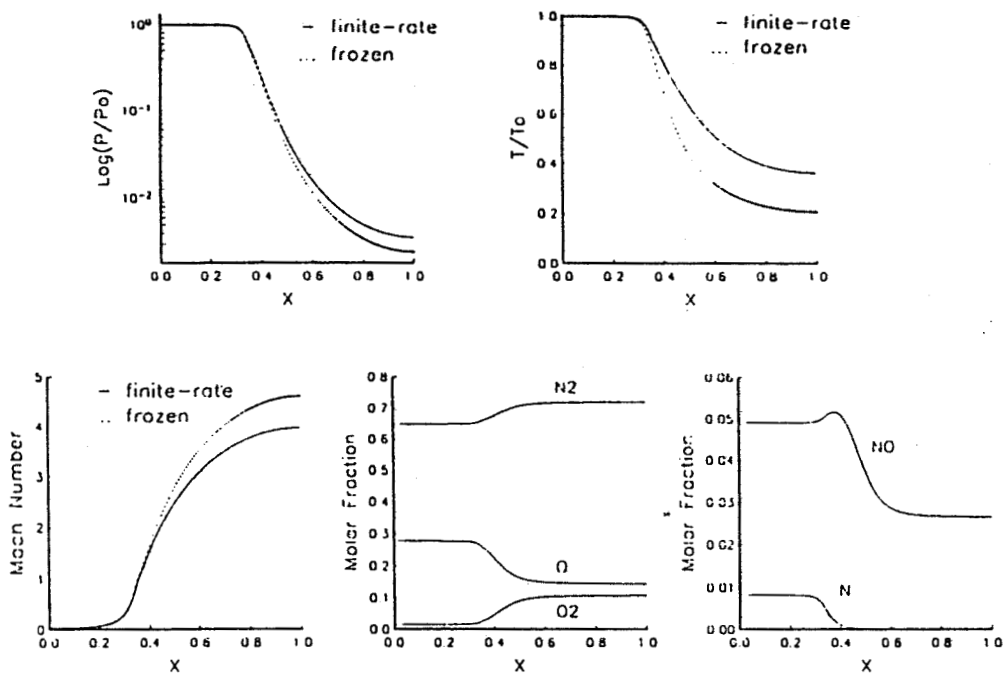
Magnitude of the Maximum Eigenvalue of the Amplification Matrix G, Pressure-Based Scheme, CFL=100.



Magnitude of the Maximum Eigenvalue of the Amplification Matrix G, Density-Based Scheme, CFL=100.

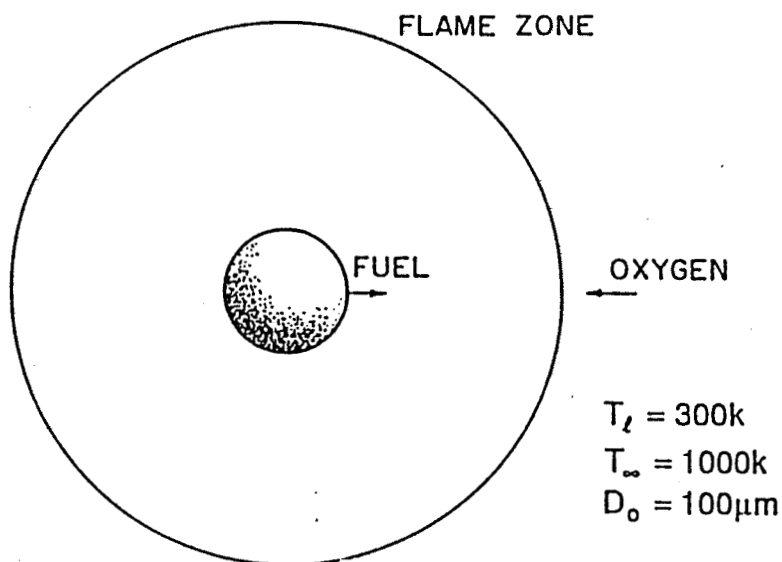
CONVERGENT-DIVERGENT NOZZLE FLOW, $M_i=0.01$

Non-Equilibrium Air Dissociation/Recombination Chemistry
5 Species, 11 Reaction Steps



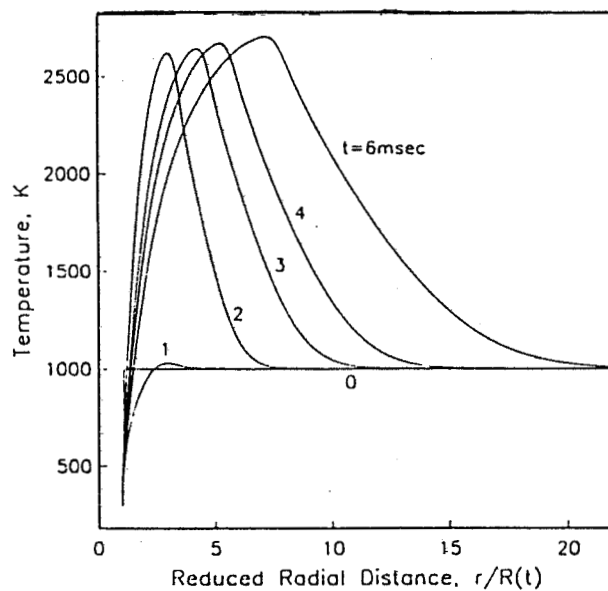
COMBUSTION OF n-PENTANE FUEL DROPLET

Non-Equilibrium n-Pentane - Air Combustion Chemistry
5 Species, 1 Global Reaction Step

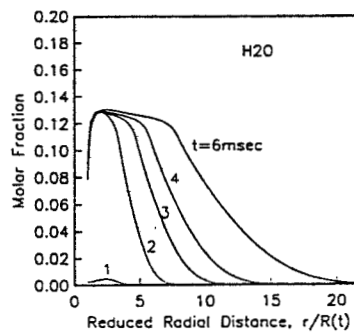
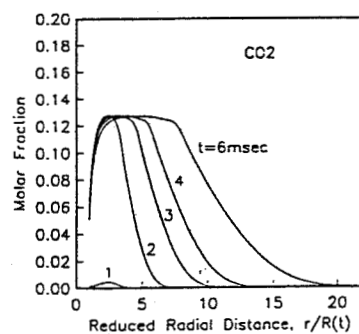
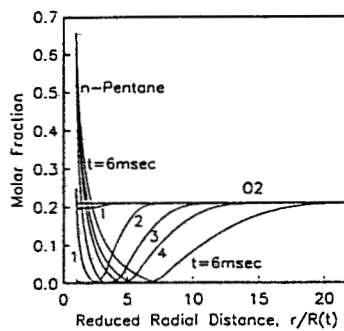


DROPLET COMBUSTION, $M \simeq 10^{-4} - 10^{-6}$

Non-Equilibrium n-Pentane - Air Combustion Chemistry
5 Species, 1 Global Reaction Step



DROPLET COMBUSTION, $M \simeq 10^{-4} - 10^{-6}$



CONCLUSIONS

- An efficient and robust numerical algorithm has been developed for chemically reacting flows at **all speeds**.

FUTURE PLAN

1. Development and validation of the 2D/3D all-speed code for reacting flows.
2. Implementation of zonal capability.
3. Implementation of turbulence, turbulence/reaction closure, spray combustion, and thermal radiation models.
4. Reacting flow simulation for gas turbine combustor, rocket motor combustion chamber, etc.

Study of Shock-Induced Combustion Using an Implicit TVD Scheme

Shayne Yungster
Institute for Computational Mechanics in Propulsion
NASA Lewis Research Center

The supersonic combustion flowfields associated with various hypersonic propulsion systems, such as the ram accelerator (a ramjet-in-tube concept), the oblique detonation wave engine, and the scramjet, are being investigated using a new CFD code. The code solves the fully coupled Reynolds-averaged Navier-Stokes equations and species continuity equations in an efficient manner. It employs an iterative method that is based on the lower-upper symmetric successive overrelaxation (LU-SSOR) implicit factorization scheme, and a second order symmetric total variation diminishing (TVD) differencing scheme. To accelerate the convergence of the basic iterative procedure, this code can be combined with vector extrapolation methods, such as the Minimal Polynomial (MPE) and the Reduced Rank (RRE) Extrapolation. The extrapolation procedure solves a linear least squares problem and produces a sequence of approximations that, in general, has better convergence properties than the sequence obtained from the iterative scheme alone. Two different formulations of the LU-SSOR factorization scheme are currently implemented. In one formulation, the implicit operator includes the full Jacobian matrix of the chemical source term, leading to a preconditioner matrix of size $n_s \times n_s$, where n_s is the number of species, which has to be inverted at every grid point. If the number of species considered is large, inverting this preconditioner can be very expensive. Therefore, a second formulation has been introduced in which the Jacobian matrix is replaced by a diagonal matrix that is designed to approximate the time scaling effects obtained by using the full Jacobian. In this formulation, no matrix inversions are required. Figure 1 shows the density residual history obtained with the two formulations (with and without extrapolation) for the case of a supersonic flow past a compression corner. The chemical nonequilibrium processes are simulated by means of a 9-species, 18-step finite-rate H_2 -air combustion model. When extrapolation is used, it is started after N_0 iterations, and is implemented in the so called "cycling" mode, using a sequence of K_{max} vectors obtained from the iterative scheme. The overhead in CPU time due to the use of extrapolation is very small (less than 1%) in the present case. The results indicate that savings of up to 40% in the overall computational work required to reach convergence can be realized by using RRE in combination with the basic iterative scheme (using both formulations). Similar results are obtained with MPE. The results also indicate that for the present chemistry model, the diagonal formulation is less efficient, requiring approximately 45% more CPU time than the Full Jacobian formulation to reach convergence. The code is currently being applied to study shock wave/boundary layer interactions in premixed combustible gases, and to investigate the ram accelerator concept. Results obtained for a ram accelerator configuration (Fig. 2) indicate a new combustion mechanism in which a shock wave induces combustion in the boundary layer, which then propagates outwards and downstream. The combustion process creates a high pressure region over the back of the projectile resulting in a net positive thrust force.

Objective

- To investigate shock-wave/boundary layer interactions involving premixed combustible gases.
- To compute (based on the Navier-Stokes equations) the supersonic combustion flowfield in a ram accelerator, a ramjet-in-tube concept.

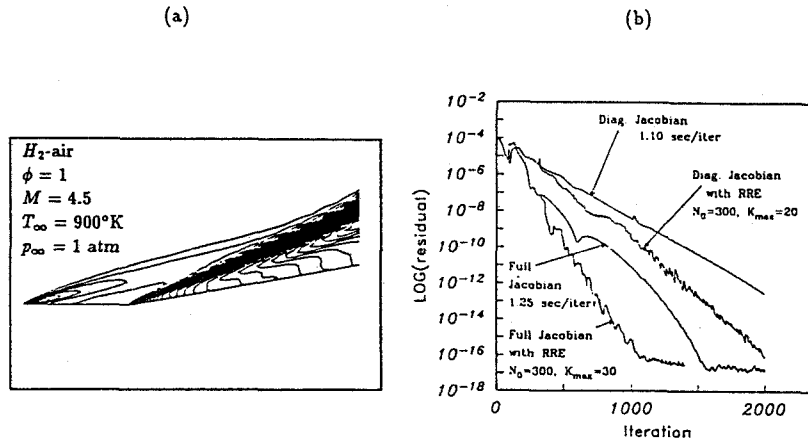


Figure 1: Supersonic reacting flow past a compression corner; (a) pressure contours; (b) Convergence history of L_2 density residual (grid, 80×50).

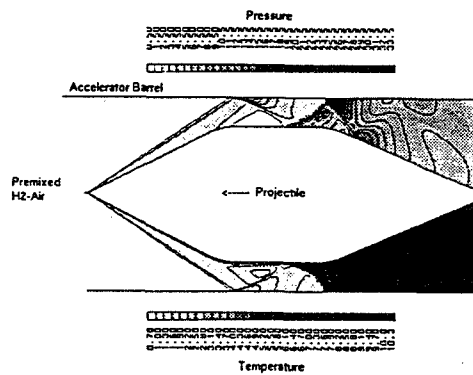


Figure 2: Nondimensional pressure (top) and temperature (bottom) contours for a ram accelerator, a ramjet-in-tube device for accelerating projectiles to ultrahigh velocities. Mixture: stoichiometric H_2 -air, $M = 6.7$, $P_\infty = 1 \text{ atm}$, $T_\infty = 300^\circ K$, $T_w = 600^\circ K$. The vertical direction is magnified by a factor of 2.

Equations and Transport properties

- 2D/axisymmetric Reynolds-Averaged Navier-Stokes equations (general coordinates).
- Species specific heat, conductivity and viscosity obtained from fourth order polynomials of temperature.
- Conductivity and viscosity of mixture - Wilke's mixing rule.
- Binary mass diffusivities - Chapman-Enskog theory.

Combustion and Turbulence Models

- 9-species, 18-step chemistry model for hydrogen-oxygen combustion.
- Baldwin-Lomax Turbulence Model

Numerical Method

- Finite difference method.
- LU-SSOR implicit factorization scheme.
- Second-order symmetric TVD differencing scheme.
- Use of extrapolation techniques for convergence acceleration (i.e. Reduced Rank and Minimal Polynomial Extrapolation).

Reduced Rank Extrapolation Method (RRE)

- Given the vector sequence $\mathbf{Q}^0, \mathbf{Q}^1, \dots, \mathbf{Q}^{k+1}$, with $k = K_{MAX}$, compute the differences

$$\Delta \mathbf{Q}^j = \mathbf{Q}^{j+1} - \mathbf{Q}^j, \quad j = 0, 1, \dots, k. \quad (1)$$

- Next, determine the scalars $\gamma_0, \gamma_1, \dots, \gamma_k$ by solving the constrained least-squares problem

$$\text{minimize} \left\| \sum_{j=0}^k \gamma_j \Delta \mathbf{Q}^j \right\| \quad (2)$$

subject to $\sum_{j=0}^k \gamma_j = 1$.

- Finally, set

$$\mathbf{S}_{0,k} = \sum_{j=0}^k \gamma_j \mathbf{Q}^j \quad (3)$$

Although the original definition of RRE is different, the definition given above is equivalent to it and results in an implementation that is more stable numerically.

Numerical Method

- LU-SSOR implicit factorization scheme:

$$LT^{-1}U\delta Q = -\Delta t[RHS]$$

$$L = I + \beta\Delta t[D_{\xi}^{-}A^{+} + D_{\eta}^{-}B^{+} - A^{-} - B^{-} - C]$$

$$U = I + \beta\Delta t[D_{\xi}^{+}A^{-} + D_{\eta}^{+}B^{-} + A^{+} + B^{+}]$$

$$T = I + \beta\Delta t[A^{+} + B^{+} - A^{-} - B^{-}]$$

- Second-order symmetric TVD differencing scheme:

$$[RHS] = (F_{j+\frac{1}{2},k} - F_{j-\frac{1}{2},k} + G_{j,k+\frac{1}{2}} - G_{j,k-\frac{1}{2}}) + W_{j,k} \\ + (R_{j+\frac{1}{2}}\Phi_{j+\frac{1}{2}} - R_{j-\frac{1}{2}}\Phi_{j-\frac{1}{2}} + R_{k+\frac{1}{2}}\Phi_{k+\frac{1}{2}} - R_{k-\frac{1}{2}}\Phi_{k-\frac{1}{2}})$$

Where

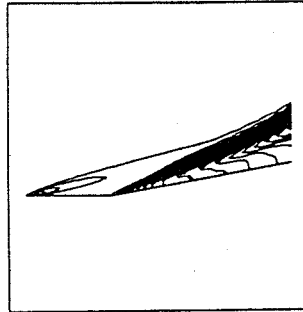
$$\phi_{j+\frac{1}{2}}^l = -\frac{1}{2}\psi(a_{j+\frac{1}{2}}^l)(\alpha_{j+\frac{1}{2}}^l - \bar{Q}_{j+\frac{1}{2}}^l)$$

$$\psi(z) = \begin{cases} |z| & \text{if } |z| \geq \delta_1 \\ (z^2 + \delta_1^2)/2\delta_1 & \text{if } |z| < \delta_1 \end{cases}$$

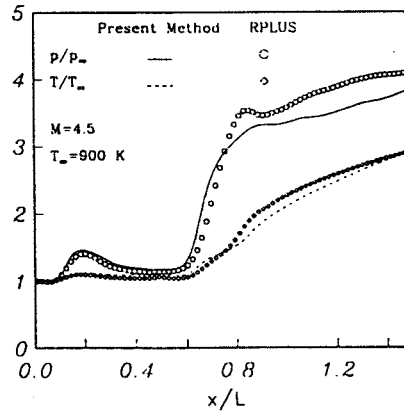
$$(\delta_1)_{j+\frac{1}{2}} = \bar{\delta}[|U_{j+\frac{1}{2}}| + |V_{j+\frac{1}{2}}| + \frac{1}{2}(\sqrt{\xi_x^2 + \xi_y^2} + \sqrt{\eta_x^2 + \eta_y^2})a_{j+\frac{1}{2}}]$$

$$.1 < \bar{\delta} < .4$$

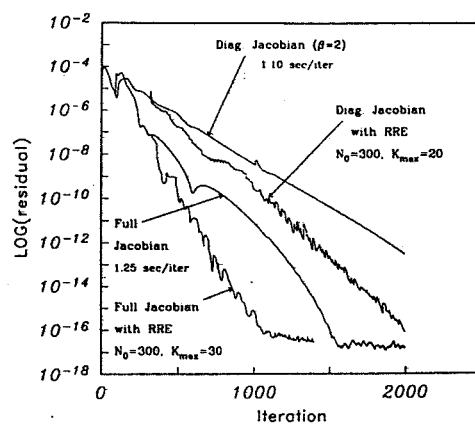
Reacting flow past a compression corner



$H_2 - Air$
 $\Phi = 1$
 $p_{\infty} = 1 \text{ atm}$
 $T_{\infty} = 900^\circ \text{ K}$
 $M = 4.5$

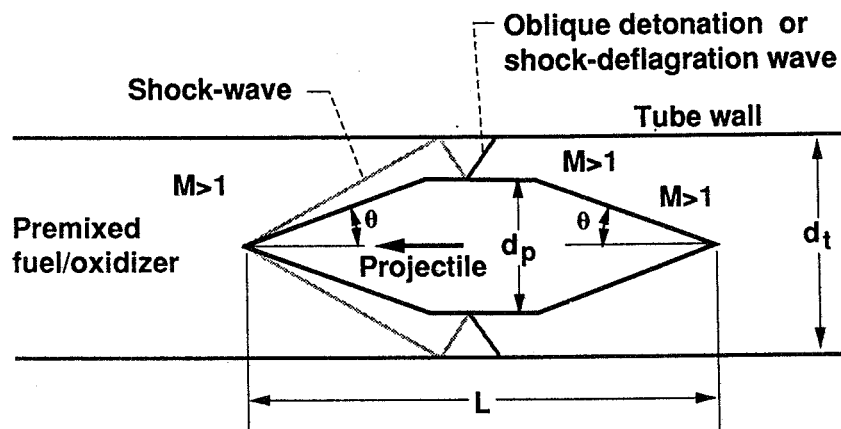


Convergence History of L_2 Density Residual



Schematic of "Oblique Detonation" Ram Accelerator

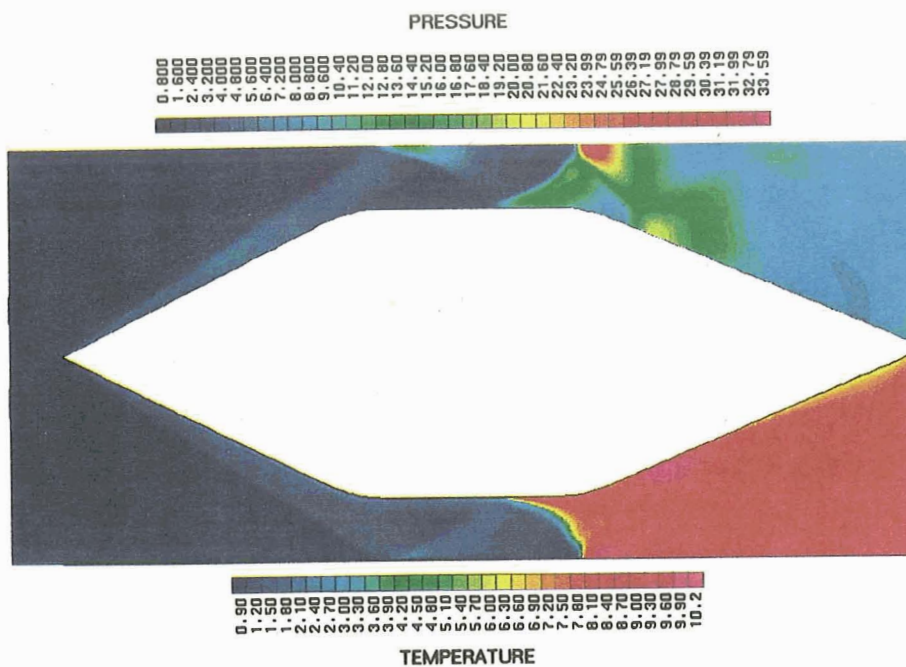
($L = 15$ cm; $d_p = 2.5$ cm; $d_t = 3.8$ cm; $\theta = 14^\circ$)



CD-91-51910

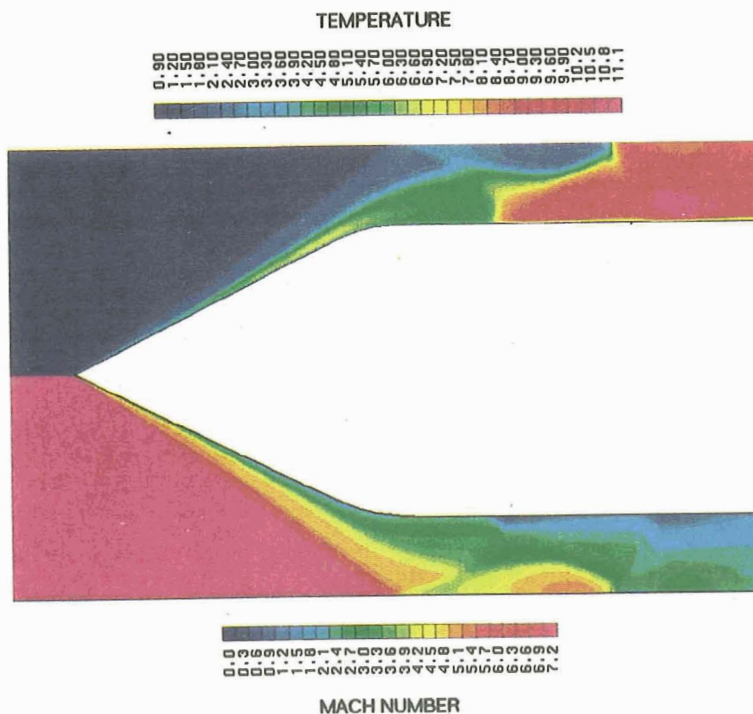
NONDIMENSIONAL PRESSURE (TOP) AND TEMPERATURE (BOTTOM) CONTOURS

Stoichiometric Hydrogen-Air; $M=6.7$; $P=1$ atm; $T=300$ K.



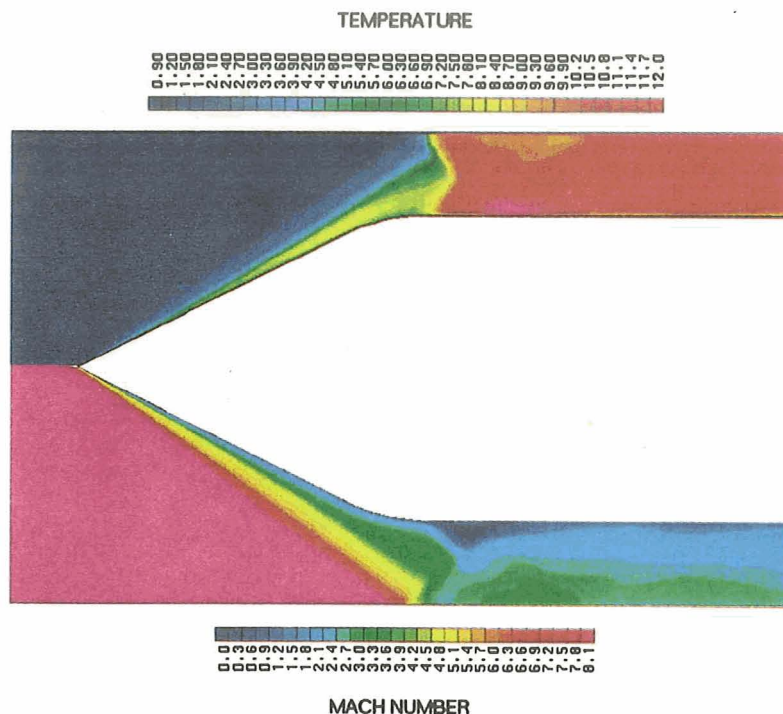
NONDIMENSIONAL TEMPERATURE (TOP) AND MACH NUMBER (BOTTOM) CONTOURS

Stoichiometric Hydrogen-Air; $M=7.2$; $P=1$ atm; $T=300$ K.



NONDIMENSIONAL TEMPERATURE (TOP) AND MACH NUMBER (BOTTOM) CONTOURS

Stoichiometric Hydrogen-Air; $M=8.0$; $P=1$ atm; $T=300$ K.



Conclusions

- A new CFD code that efficiently solves the Navier-Stokes equations with finite-rate chemistry was presented.
- The application of vector extrapolation methods to viscous, chemically reacting flows was demonstrated.
- Results indicate that significant savings in computational work can be realized by using vector extrapolation methods in combination with the basic iterative scheme.
- Ram Accelerator Concept: results indicate that viscous effects are of primary importance. The combustion processes in the boundary layer strongly affect the entire flowfield.

ORIGINAL PAGE
COLOR PHOTOGRAPH

UPWIND SCHEMES AND BIFURCATING SOLUTIONS
IN REAL GAS COMPUTATIONS

Ambady Suresh
Sverdrup Technology, Inc.
Lewis Research Center Group
NASA Lewis Research Center
and
Meng-Sing Liou
Internal Fluid Mechanics Division
NASA Lewis Research Center

The area of high speed flow is seeing a renewal of interest due to advanced propulsion concepts such as the NASP, Space Shuttle, and future civil transport concepts. Upwind schemes to solve such flows have become increasingly popular in the last decade due mainly to their excellent shock capturing properties. Within this class of upwind schemes the Osher scheme has a few distinct advantages, such as a continuously differentiable numerical flux (making it especially attractive for implicit schemes), entropy satisfying solutions and the exact resolution of stationary shocks and contacts.

High speed flow is generally accompanied by changes in the chemical and thermodynamic composition of the gas which need to be taken into account. In the first part of this paper we present the extension of the Osher scheme to equilibrium and non-equilibrium gases. For the equilibrium case, the extension proceeds by choosing the pressure and entropy as the independent thermodynamic variables so that the Riemann invariants can be reduced to quadratures. Efficiency is achieved by noting that the quadrature need be only as accurate as the discretization error. In the non-equilibrium case, the determination of the intermediate points using Riemann invariants involves iterative solution of a transcendental equation. As this is computationally inefficient, an approximate method which involves no iteration is presented. For simplicity, the source terms are treated explicitly. In both cases, the formal extension is unique unlike the Roe scheme where there is a one parameter family of solutions with no clear choice among them.

Computations based on the above scheme are presented to demonstrate the feasibility, accuracy and efficiency of the proposed scheme. One of the test problems is a Chapman-Jouguet detonation problem for which numerical solutions have been known to bifurcate into spurious weak detonation solutions on coarse grids. Our results indicate that the numerical solution obtained depends both on the upwinding scheme used and the limiter employed to obtain second order accuracy. For example, the Osher scheme gives the correct CJ solution when the super-bee limiter is used but gives the spurious solution when the Van Leer limiter is used. With the Roe scheme the spurious solution is obtained for all limiters.

OVERVIEW

- Extension of the Osher Scheme for reacting flows.
 - 2 species system
 - Intermediate and Sonic points
 - Numerical results
- Bifurcating $Z - N - D$ Detonations.
 - Theory
 - Numerical Results

THEORY [Colella, Majda, Royturd (86)]

Simplified Model:

$$u_t + \left(\frac{1}{2} u^2 + \Delta h z \right)_\eta = 0$$

$$u \sim p/\rho$$

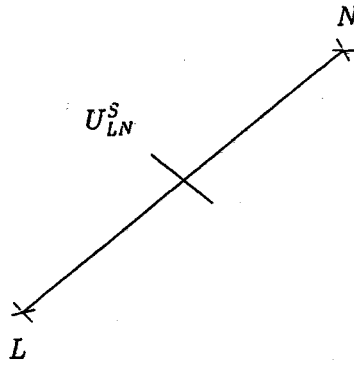
$$z_n = k\phi(u)z$$

For large $k\Delta x$ and/or Δh non-physical weak detonation waves exist which travel at mesh speed $\frac{\Delta x}{\Delta t}$.

NUMERICAL RESULTS

- Riemann Problems (ideal gas)
- Reacting nozzle flow
- $Z - N - D$ detonations

Sonic Points



Using invariants is expensive due to exponents.

$$\tau_s = \frac{\lambda_L^1}{(\lambda_L^1 - \lambda_N^1)}$$

$$U_{LN}^S = U_L(1 - \tau_s) + U_R\tau_s$$

Freeze eigenvectors at $(U_L + U_R)/2$ and evaluate wave strengths as:

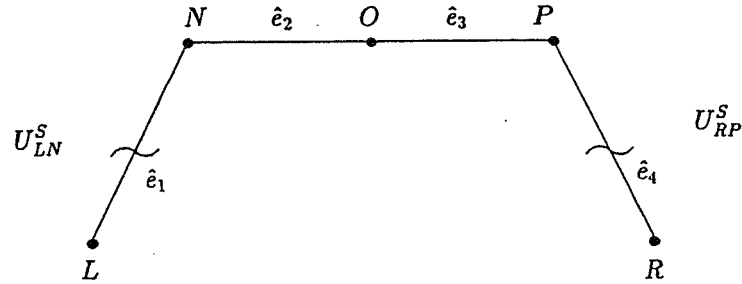
$$\Delta\tau_{1,4} = \frac{1}{2a^2} [\Delta p \mp pa\Delta u]$$

$$\Delta\tau_2 = \frac{1}{a^2} [-\Delta p + a^2\Delta p]$$

$$\Delta\tau_3 = \Delta z$$

Intermediate states obtained by translation from left and right.

HRP



For multispecies U_N and U_P can be obtained only iteratively:

$$a p_N^\alpha + b p_N^\beta = c$$

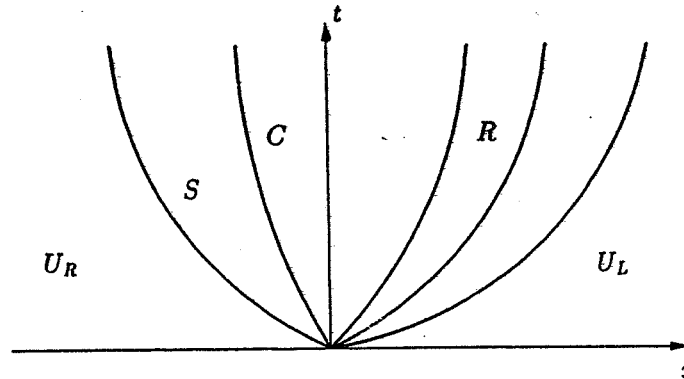
However, U_N , U_P need only be as accurate as the overall discretization error.

- 1) Source terms do not change wave pattern.
 - 2) Along $x = 0$, solution approaches homogeneous Riemann problem near origin. Li and Yu [1985]
- STRATEGY

Use homogeneous Riemann problem [Gottlieb, etc.] for upwinding and add in source terms later.

Riemann Problem: no longer self-similar.

$$R(x, t, U_L, U_R) \neq R\left(\frac{x}{t}, U_L, U_R\right)$$



Interface flux $F_{i+\frac{1}{2}}$ not well defined.

However . . .

2 Species System:

$$\begin{bmatrix} \rho \\ \rho u \\ \rho E \\ \rho z \end{bmatrix}_t + \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho E u + p u \\ \rho u z \end{bmatrix}_x = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -\rho k \end{bmatrix}$$

z = mass fraction. $z = 1$ (unburnt) $z = 0$ (burnt)

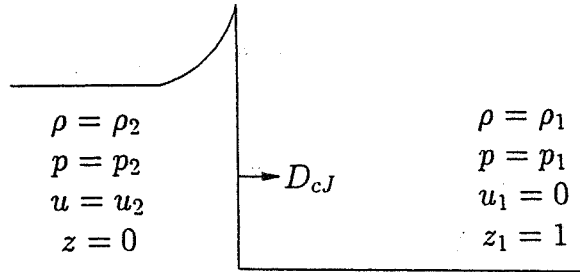
$k = k(z, T)$ = reaction rate

$$E = \frac{p}{(\gamma(z)-1)\rho} + \frac{1}{2}u^2 - z\Delta h$$

$$\gamma(z) = \frac{\gamma_1 C_{V1} z + \gamma_2 C_{V2} (1-z)}{C_{V1} z + C_{V2} (1-z)}$$

BIFURCATING Z - N - D DETONATIONS

Problem



$$r(z) = \text{constant.}$$

Solved on moving grid.

k - Step function of T .

NUMERICAL RESULTS

- Bifurcation depends on limiter used and Riemann Solver.

R. S. \ Limiter	1 st Order	L.L. Limiter	Superbee
Osher-N	B	B	NB
Osher-R	B	B	NB
Roe	B	B	B

IDEAL GAS SHOCK TUBE PROBLEM

OSHER SCHEME NATURAL ORDER

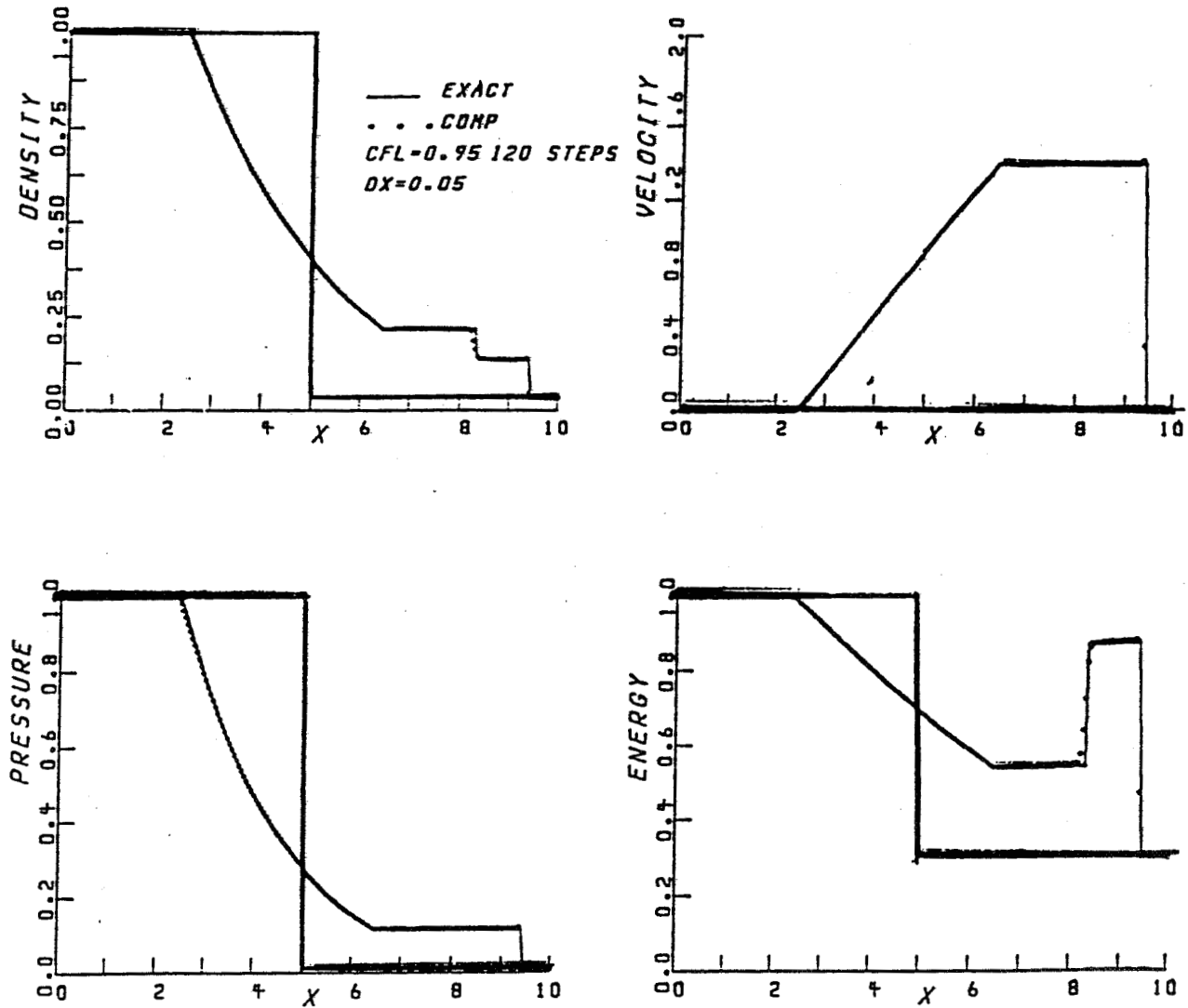


Fig. 2: Ideal gas shock tube problem. Frozen solution.

C - J DETONATION PROBLEM

OSHER SCHEME NATURAL ORDER

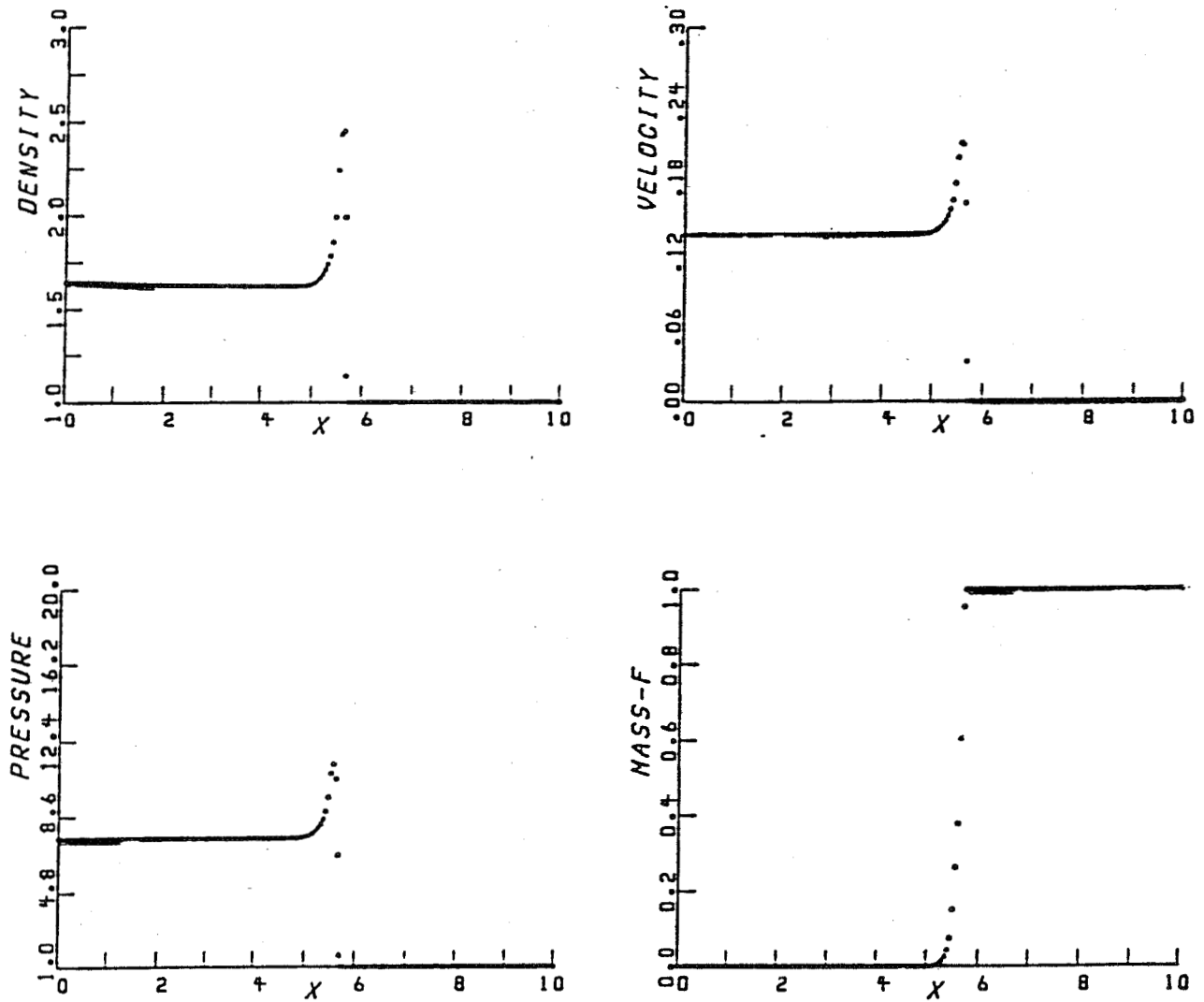


Fig. 3: Chapman-Jouguet detonation problem on a fine grid.
Reaction zone = 10 cells

C - J DETONATION PROBLEM

OSHER SCHEME NATURAL ORDER

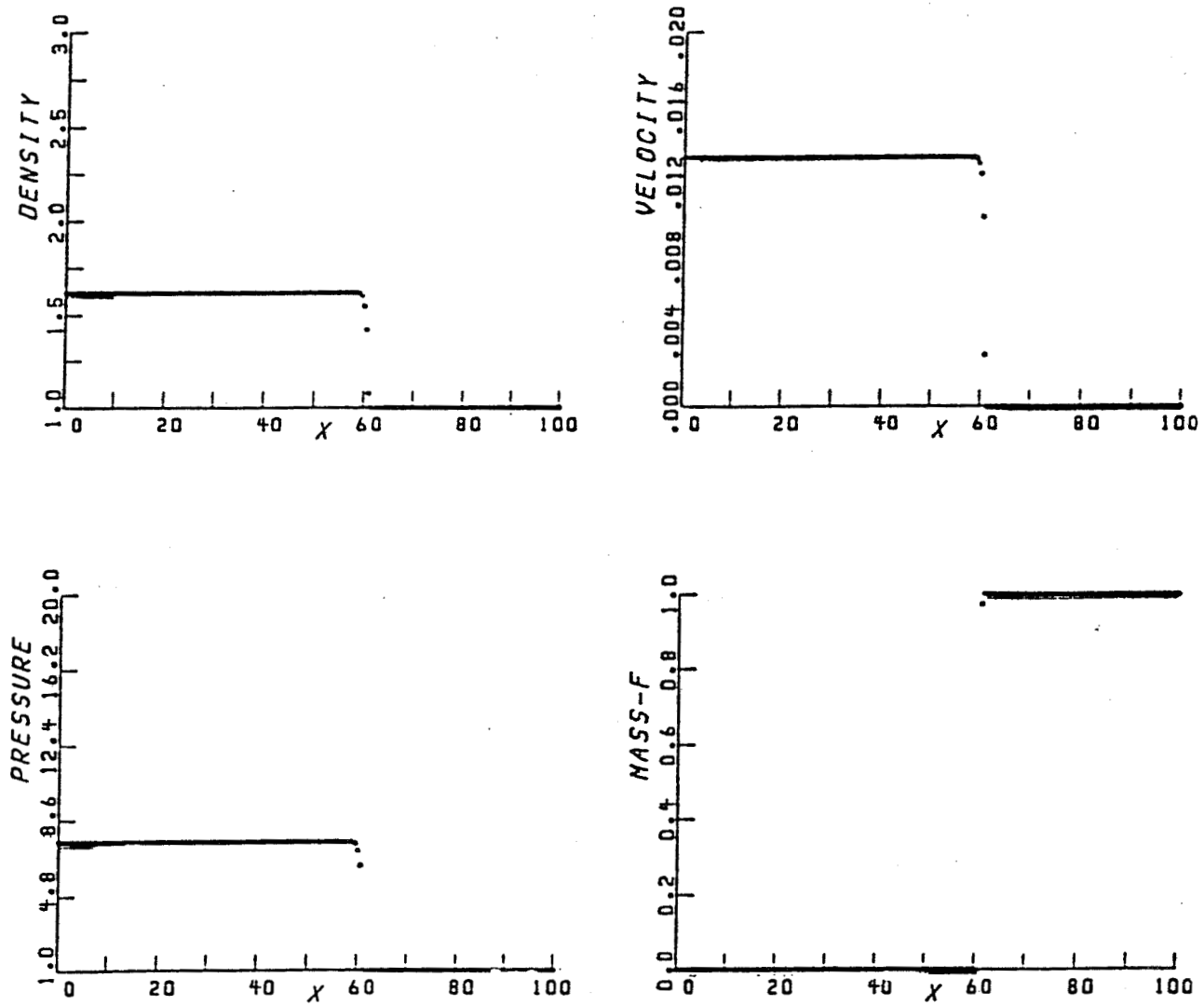


Fig. 4a: Chapman-Jouquet detonation problem on a coarse grid.
Reaction zone = 1/10 cell, Superbee Limiter.

C - J DETONATION PROBLEM

OSHER SCHEME NATURAL ORDER

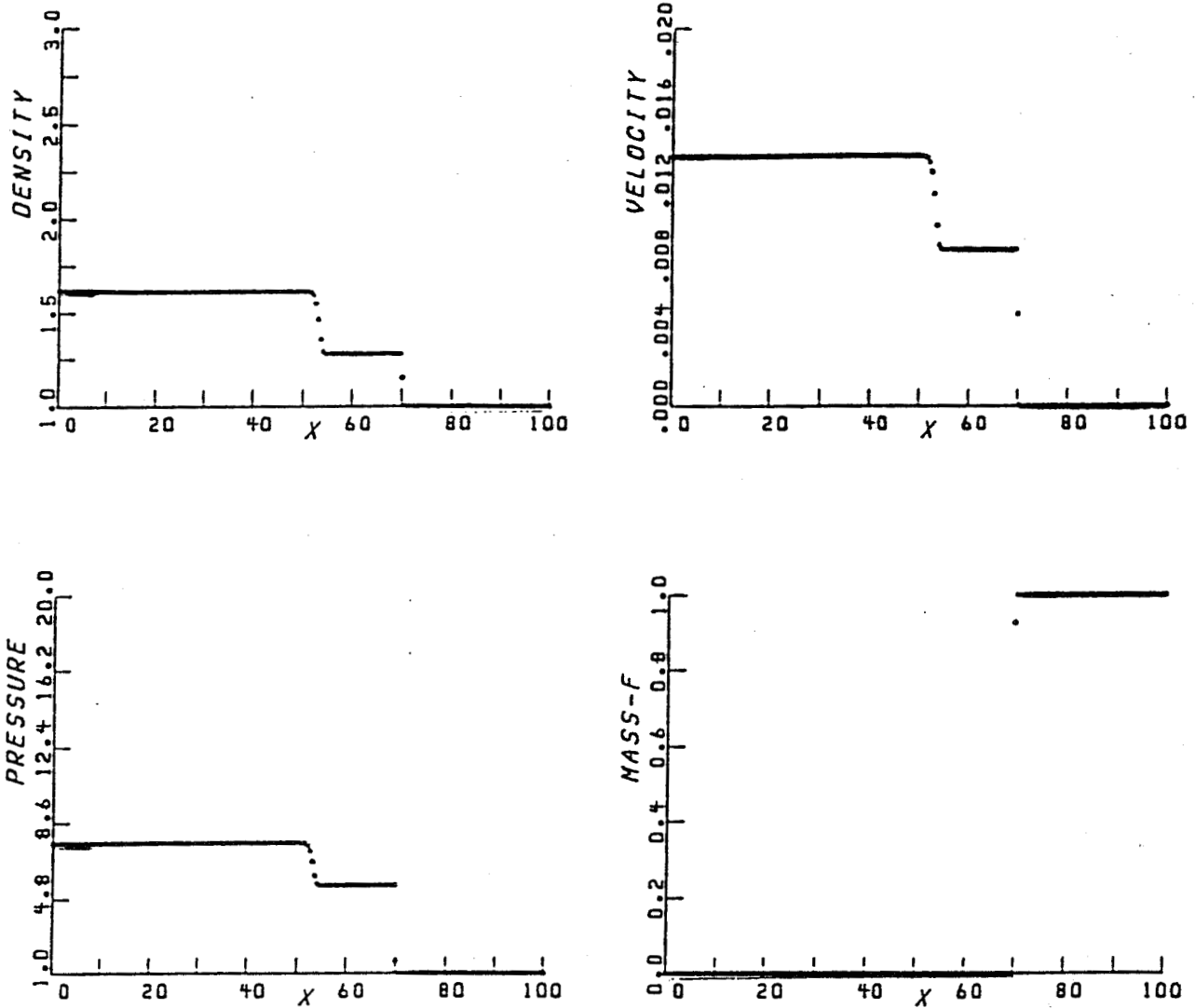


Fig. 4b: Chapman-Jouguet detonation problem on a coarse grid.
Reaction zone = 1/10 cell, Vanleer Limiter.

DIVERGENT NOZZLE
OSHER SCHEME NATURAL ORDER

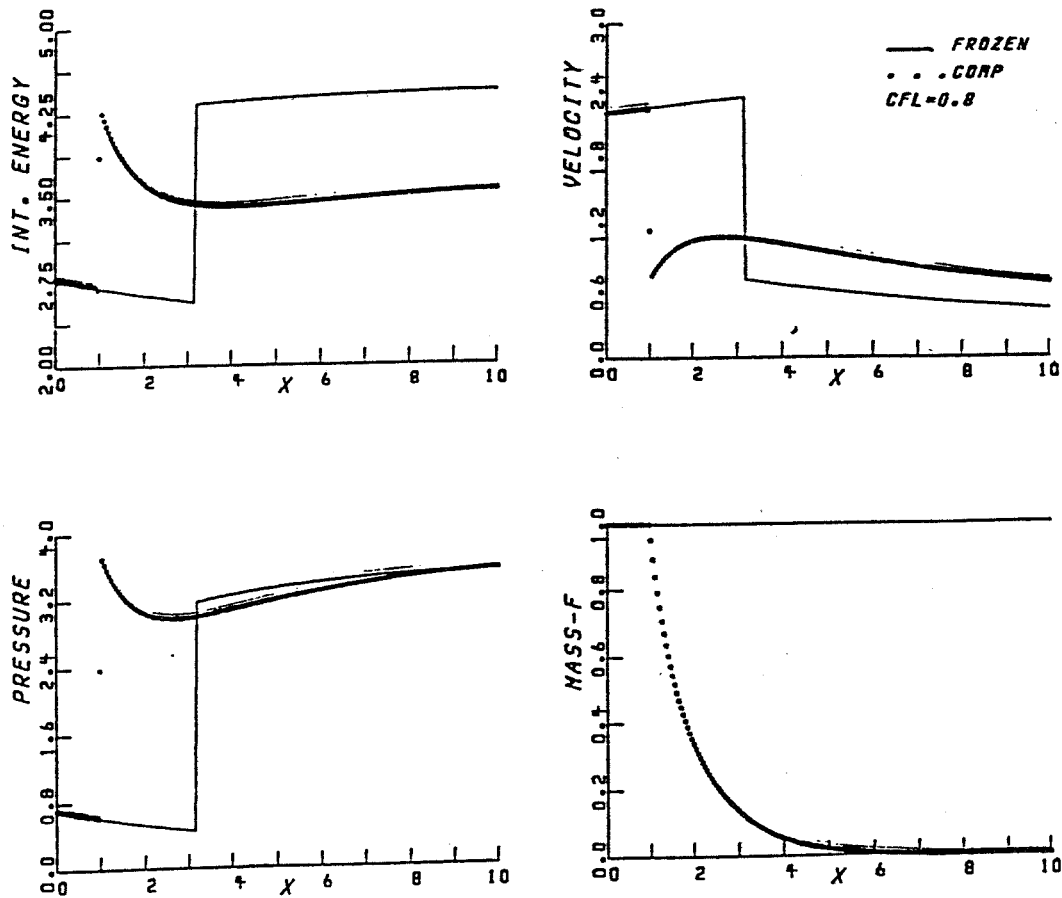


Fig. 5: Divergent Nozzle Problem. Reaction zone = 20 cells.

CONCLUSIONS

- The Osher scheme can be extended efficiently to reacting flows.
- Bifurcations in reacting flows require further study.

HIGH ORDER PARALLEL NUMERICAL SCHEMES FOR SOLVING INCOMPRESSIBLE FLOWS

Avi Lin
University of Pennsylvania

Edward J. Milner, May-Fun Liou, and Richard A. Belch
Internal Fluid Mechanics Division
NASA Lewis Research Center

The use of parallel computers for numerically solving flow fields has gained a lot of importance in recent years. In most cases, the parallel machine executes modified standard serial CFD codes, taking advantage of parallelism by using special constructs, directives, system commands and calls. In other instances, numerical schemes and computational algorithms have been changed or redesigned to increase performance of the overall algorithm. This paper presents a new high order numerical scheme for CFD specifically designed for parallel computational environments.

A distributed MIMD system gives the flexibility of treating different elements of the governing equations with totally different numerical schemes in different regions of the flow field. This heterogeneous parallel numerical approach is sometimes technically complicated in terms of programming and administering the various processes, but it is quite efficient and contains enough free parameters for optimizing execution speedup.

The parallel decomposition of the governing operator to be solved is the primary parallel split. The decomposition of the physical domain of the flow into subdomains is the secondary, or optional, split. At the coarsest parallel level, each of the processors is assigned to solve a suboperator of the original PDE operator over a given subdomain. An iterative numerical procedure is employed. All of these splittings are designed to ensure a high rate of convergence.

The primary parallel split was studied using a hypercube-like architecture having clusters of shared-memory processors at each node. The approach is demonstrated using as examples some simple steady-state incompressible flows. Future studies should investigate the secondary split because, depending on the numerical scheme that each of the processors applies and the nature of the flow in the specific subdomain, it may be possible for a processor to seek a better, or higher order, scheme for its particular subcase.

OBJECTIVE:

TO REDUCE THE TIME REQUIRED TO SOLVE
CFD PROBLEMS BY USING PARALLEL
PROCESSING TECHNIQUES

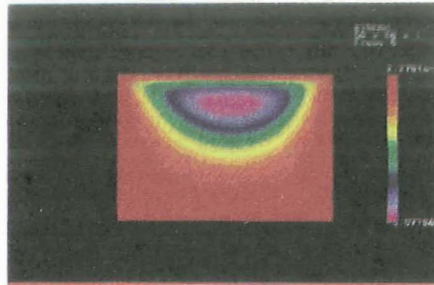
COMPUTATIONAL TECHNOLOGIES BRANCH

DISCUSSION FRAMEWORK

- PARALLEL PROCESSING TECHNIQUE FOR
SOLVING A BLOCK TRIDIAGONAL SYSTEM
- SOME APPLICATIONS & RESULTS
- PARALLEL PROCESSING TECHNIQUE FOR
INVERTING A MATRIX

COMPUTATIONAL TECHNOLOGIES BRANCH

A Parallel, Scalable, 2D Navier-Stokes Solver



Stream function, driven cavity, 50 by 50 grid

- Parallel algorithm design provides efficient implementation on distributed or shared memory machines
 - 74% efficiency achieved on Hypercluster test bed
- Scalable and portable algorithm
 - basic building blocks (e.g. matrix solvers)
 - total solver
- Second-order accurate

NASA

LEWIS RESEARCH CENTER

ASSUME THAT

$$1) \quad B_{m,1} = K_{m,1} * B_{1,1}$$

$$2) \quad B_{m,2} = K_{m,2} * B_{2,2}$$

$$3) \quad B_{m,3} = K_{m,3} * B_{3,3}$$

COMPUTATIONAL TECHNOLOGIES BRANCH

THEN

- 1) $(A_{1,1} + A_{1,2} * K_{2,1} + A_{1,3} * K_{3,1}) * B_{1,1} = I$
- 2) $(A_{2,1} * K_{1,2} + A_{2,2} + A_{2,3} * K_{3,2}) * B_{2,2} = I$
- 3) $(A_{3,1} * K_{1,3} + A_{3,2} * K_{2,3} + A_{3,3}) * B_{3,3} = I$
- 4) $A_{1,1} * \textcircled{K_{1,2}} + A_{1,2} + A_{1,3} * K_{3,2} = 0$
- 5) $A_{2,1} * K_{1,3} + A_{2,2} * \textcircled{K_{2,3}} + A_{2,3} = 0$
- 6) $A_{3,1} + A_{3,2} * K_{2,1} + A_{3,3} * \textcircled{K_{3,1}} = 0$
- 7) $A_{1,1} * \boxed{K_{1,3}} + A_{1,2} * K_{2,3} + A_{1,3} = 0$
- 8) $A_{2,1} + A_{2,2} * \boxed{K_{2,1}} + A_{2,3} * K_{3,1} = 0$
- 9) $A_{3,1} * K_{1,2} + A_{3,2} + A_{3,3} * \boxed{K_{3,2}} = 0$

COMPUTATIONAL TECHNOLOGIES BRANCH

THUS,

- 1) $B_{1,1} = (A_{1,1} + A_{1,2} * \boxed{K_{2,1}} + A_{1,3} * \textcircled{K_{3,1}})^{-1}$
- 2) $B_{2,2} = (A_{2,1} * \textcircled{K_{1,2}} + A_{2,2} + A_{2,3} * \boxed{K_{3,2}})^{-1}$
- 3) $B_{3,3} = (A_{3,1} * \boxed{K_{1,3}} + A_{3,2} * \textcircled{K_{2,3}} + A_{3,3})^{-1}$
- 4) $\textcircled{K_{1,2}} = -A_{1,1}^{-1} * (A_{1,2} + A_{1,3} * \boxed{K_{3,2}})$
- 5) $\textcircled{K_{2,3}} = -A_{2,2}^{-1} * (A_{2,1} * \boxed{K_{1,3}} + A_{2,3})$
- 6) $\textcircled{K_{3,1}} = -A_{3,3}^{-1} * (A_{3,1} + A_{3,2} * \boxed{K_{2,1}})$
- 7) $\boxed{K_{1,3}} = (A_{1,1} - A_{1,2} * A_{2,2}^{-1} * A_{2,1})^{-1} * (A_{1,2} * A_{2,2}^{-1} * A_{2,3} - A_{1,3})$
- 8) $\boxed{K_{2,1}} = (A_{2,2} - A_{2,3} * A_{3,3}^{-1} * A_{3,2})^{-1} * (A_{2,3} * A_{3,3}^{-1} * A_{3,1} - A_{2,1})$
- 9) $\boxed{K_{3,2}} = (A_{3,3} - A_{3,1} * A_{1,1}^{-1} * A_{1,3})^{-1} * (A_{3,1} * A_{1,1}^{-1} * A_{1,2} - A_{3,2})$

COMPUTATIONAL TECHNOLOGIES BRANCH

SUMMARY

THE PARALLEL SOLVER:

+++):

- 1) IS GENERAL
- 2) CONVERGES VERY QUICKLY
- 3) IS READILY EXPANDABLE
 - a) CAN BE USED WITH ONLY ONE PROCESSOR
 - b) WILL USE AS FEW OR AS MANY PROCESSORS AS ARE AVAILABLE

---):

- 1) PERFORMS SOME REDUNDANT CALCULATION BECAUSE OF OVERLAP OF THE DATA

COMPUTATIONAL TECHNOLOGIES BRANCH

BLOCK TRIDIAGONAL SYSTEM

$$\begin{bmatrix}
 BC & & & & & & & & & \\
 ABC & & & & & & & & & \\
 & ABC & & & & & & & & \\
 & & \boxed{ABC} & & & & & & & \\
 & & & ABC & & & & & & \\
 & & & & ABC & & & & & \\
 & & & & & ABC & & & & \\
 & & & & & & \textcircled{ABC} & & & \\
 & & & & & & & ABC & & \\
 & & & & & & & & ABC & \\
 & & & & & & & & & \boxed{ABC} \\
 & & & & & & & & & \cdot \\
 & & & & & & & & & & ABC \\
 & & & & & & & & & & & ABC \\
 & & & & & & & & & & & & AB
 \end{bmatrix}
 *
 \begin{bmatrix}
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u
 \end{bmatrix}
 =
 \begin{bmatrix}
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D
 \end{bmatrix}$$

COMPUTATIONAL TECHNOLOGIES BRANCH

BLOCK TRIDIAGONAL SYSTEM

$$\begin{bmatrix}
 BC & & & & & & & & & \\
 ABC & & & & & & & & & \\
 & ABC & & & & & & & & \\
 & & \boxed{ABC} & & & & & & & \\
 & & ABC & & & & & & & \\
 & & ABC & & & & & & & \\
 & & & ABC & & & & & & \\
 \hline
 & & & \textcircled{ABC} & & & & & & \\
 & & & ABC & & & & & & \\
 & & & ABC & & & & & & \\
 & & & & ABC & & & & & \\
 & & & & & \boxed{ABC} & & & & \\
 & & & & & \bullet & & & & \\
 & & & & & ABC & & & & \\
 & & & & & ABC & & & & \\
 & & & & & AB & & & &
 \end{bmatrix}
 *
 \begin{bmatrix}
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 \bullet \\
 u \\
 u \\
 u
 \end{bmatrix}
 =
 \begin{bmatrix}
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 \bullet \\
 D \\
 D \\
 D
 \end{bmatrix}$$

COMPUTATIONAL TECHNOLOGIES BRANCH

BLOCK TRIDIAGONAL SYSTEM

$$\begin{bmatrix}
 BC & & & & & & & & & \\
 ABC & & & & & & & & & \\
 & ABC & & & & & & & & \\
 & & \boxed{ABC} & & & & & & & \\
 & & ABC & & & & & & & \\
 & & ABC & & & & & & & \\
 & & & \textcircled{ABC} & & & & & & \\
 & & & ABC & & & & & & \\
 & & & & ABC & & & & & \\
 & & & & & \boxed{ABC} & & & & \\
 & & & & & \bullet & & & & \\
 & & & & & ABC & & & & \\
 & & & & & ABC & & & & \\
 & & & & & AB & & & &
 \end{bmatrix}
 *
 \begin{bmatrix}
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 \bullet \\
 u \\
 u \\
 u
 \end{bmatrix}
 =
 \begin{bmatrix}
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 \bullet \\
 D \\
 D \\
 D
 \end{bmatrix}$$

COMPUTATIONAL TECHNOLOGIES BRANCH

BLOCK TRIDIAGONAL SYSTEM

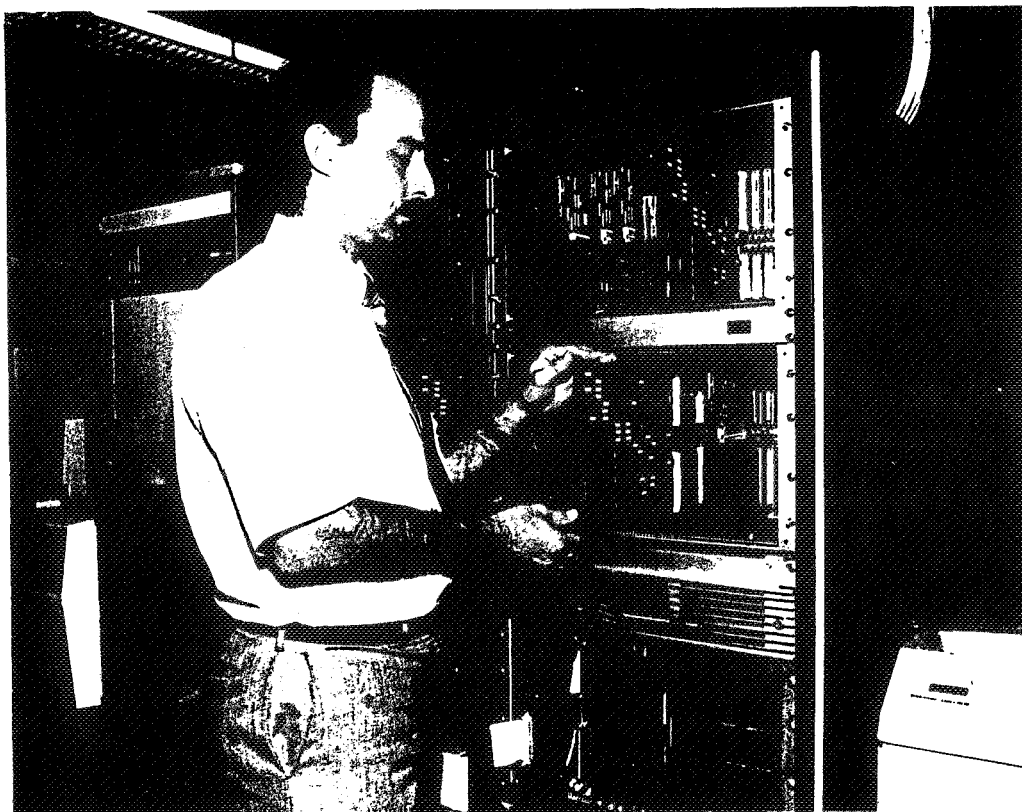
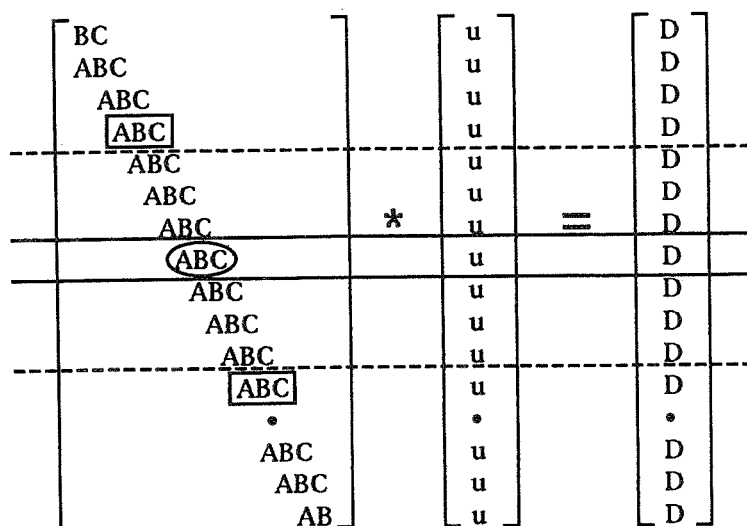
$$\begin{bmatrix}
 BC \\
 ABC \\
 ABC \\
 \boxed{ABC} \\
 ABC \\
 ABC \\
 ABC \\
 \textcircled{ABC} \\
 \hline
 ABC \\
 ABC \\
 ABC \\
 \boxed{ABC} \\
 \bullet \\
 ABC \\
 ABC \\
 AB
 \end{bmatrix}
 *
 \begin{bmatrix}
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 u \\
 \hline
 u \\
 u \\
 u \\
 u \\
 \bullet \\
 u \\
 u \\
 u
 \end{bmatrix}
 =
 \begin{bmatrix}
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 D \\
 \hline
 D \\
 D \\
 D \\
 D \\
 \bullet \\
 D \\
 D \\
 D
 \end{bmatrix}$$

COMPUTATIONAL TECHNOLOGIES BRANCH

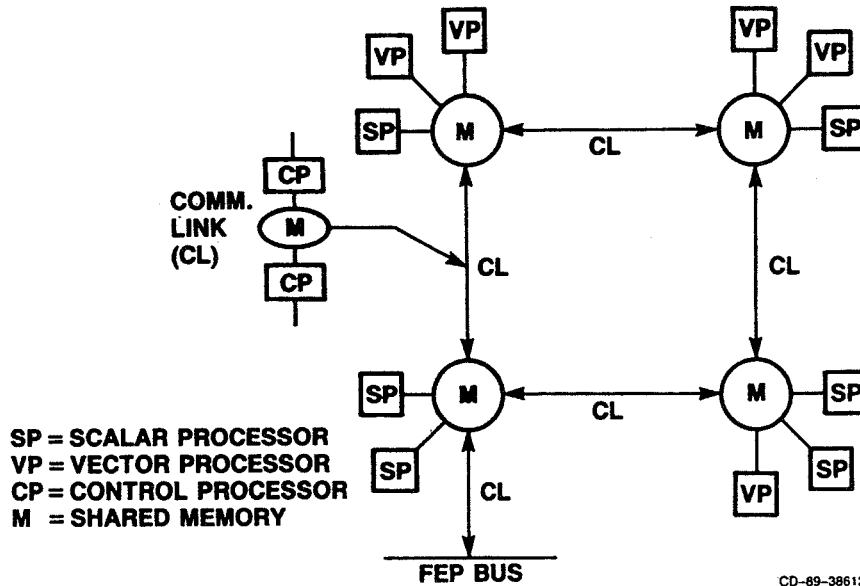
BLOCK TRIDIAGONAL SYSTEM

$$\begin{bmatrix}
 \boxed{BC} \\
 \textcircled{ABC} \\
 \boxed{AB}
 \end{bmatrix}
 *
 \begin{bmatrix}
 \boxed{u} \\
 \textcircled{u} \\
 \boxed{u}
 \end{bmatrix}
 =
 \begin{bmatrix}
 \boxed{D} \\
 \textcircled{D} \\
 \boxed{D}
 \end{bmatrix}$$

COMPUTATIONAL TECHNOLOGIES BRANCH



HYPERCLUSTER TEST BED ARCHITECTURE



CD-89-38612

NASA

LEWIS RESEARCH CENTER

FLOW NEAR A ROTATING DISK

EQUATIONS:

$$u \frac{\partial u}{\partial r} - \frac{v^2}{r} + w \frac{\partial u}{\partial z} = -\frac{1}{\rho} \frac{\partial p}{\partial r} + \nu \left\{ \frac{\partial^2 u}{\partial r^2} + \frac{\partial}{\partial r} \left(\frac{u}{r} \right) + \frac{\partial^2 u}{\partial z^2} \right\}$$

$$u \frac{\partial v}{\partial r} + \frac{uv}{r} + w \frac{\partial v}{\partial z} = \nu \left\{ \frac{\partial^2 v}{\partial r^2} + \frac{\partial}{\partial r} \left(\frac{v}{r} \right) + \frac{\partial^2 v}{\partial z^2} \right\}$$

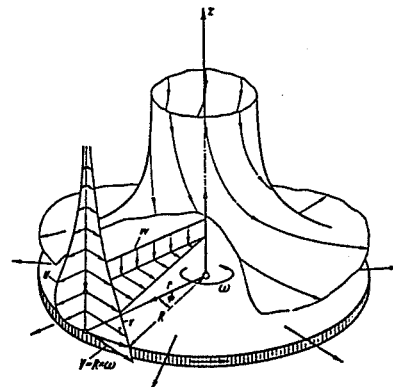
$$u \frac{\partial w}{\partial r} + w \frac{\partial w}{\partial z} = -\frac{1}{\rho} \frac{\partial p}{\partial z} + \nu \left\{ \frac{\partial^2 w}{\partial r^2} + \frac{1}{r} \frac{\partial w}{\partial r} + \frac{\partial^2 w}{\partial z^2} \right\}$$

$$\frac{\partial u}{\partial r} + \frac{u}{r} + \frac{\partial w}{\partial z} = 0.$$

ITERATION

RESIDUAL ERROR

1	MAXIMUM ERROR = 0.15019E+02
2	MAXIMUM ERROR = 0.52285E+01
3	MAXIMUM ERROR = 0.23837E+01
4	MAXIMUM ERROR = 0.10327E+01
5	MAXIMUM ERROR = 0.39308E+00
6	MAXIMUM ERROR = 0.94527E-01
7	MAXIMUM ERROR = 0.47686E-02
8	MAXIMUM ERROR = 0.14872E-04
9	MAXIMUM ERROR = 0.78692E-09
10	MAXIMUM ERROR = 0.60396E-13



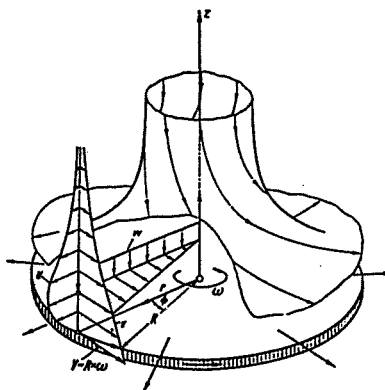
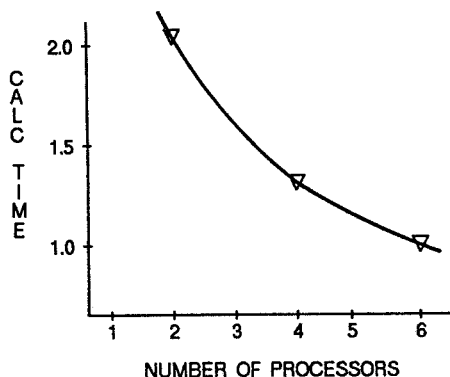
Flow in the neighborhood of a disk rotating in a fluid at rest

Velocity components: u -radial, v -circumferential, w -axial.
A layer of fluid is carried by the disk, owing to the action of viscous forces. The centrifugal forces in this layer give rise to secondary flow which is directed radially outward.

COMPUTATIONAL TECHNOLOGIES BRANCH

FLOW NEAR A ROTATING DISK

- DISTRIBUTED MEMORY SIMULATION
- SERIAL MATRIX INVERTER



Flow in the neighborhood of a disk rotating in a fluid at rest

Velocity components: u -radial, v -circumferential, w -axial. A layer of fluid is carried by the disk owing to the action of viscous forces. The centrifugal forces in this layer give rise to secondary flow which is directed radially outward.

COMPUTATIONAL TECHNOLOGIES BRANCH

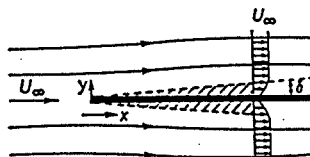
BOUNDARY LAYER ALONG A PLATE

EQUATIONS:

$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = \nu \frac{\partial^2 u}{\partial y^2},$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0,$$

$$y=0: u=v=0; \quad y=\infty: u=U_\infty$$



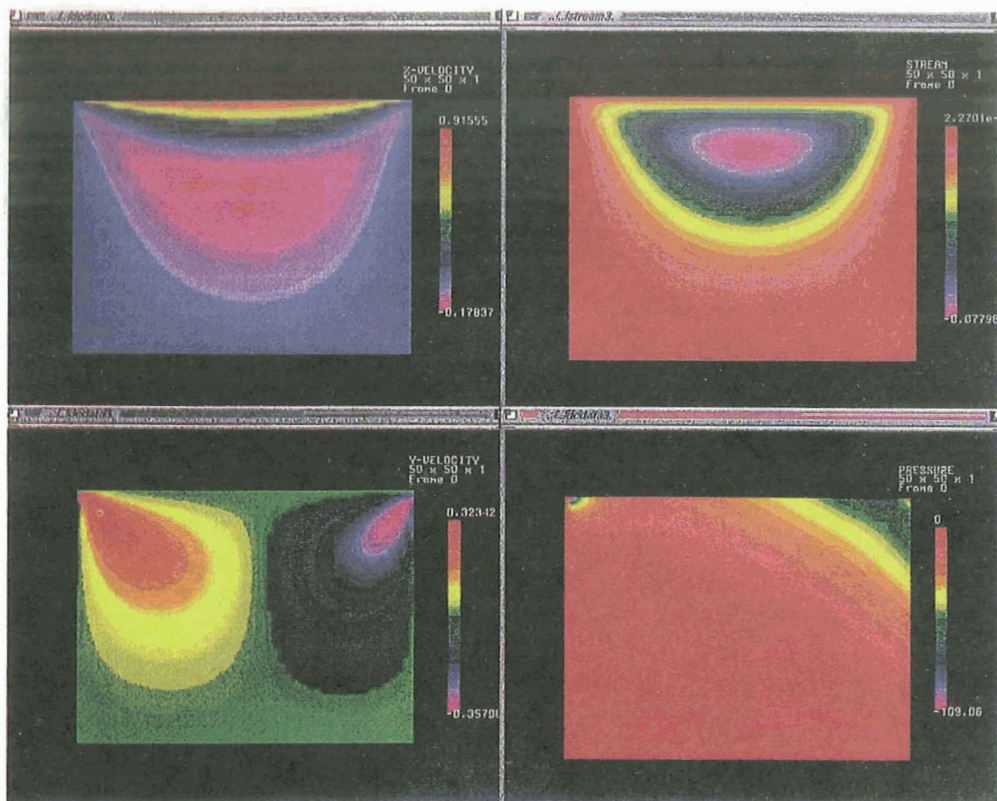
The boundary layer along a flat plate at zero incidence

ITERATION

RESIDUAL ERROR

1	MAXIMUM ERROR = 0.14258E+01
2	MAXIMUM ERROR = 0.26890E+00
3	MAXIMUM ERROR = 0.28632E-01
4	MAXIMUM ERROR = 0.24873E-03
5	MAXIMUM ERROR = 0.16154E-07
6	MAXIMUM ERROR = 0.11084E-11

COMPUTATIONAL TECHNOLOGIES BRANCH



NASA

LEWIS RESEARCH CENTER

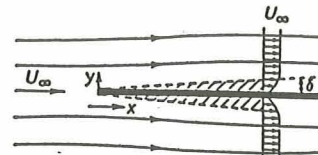
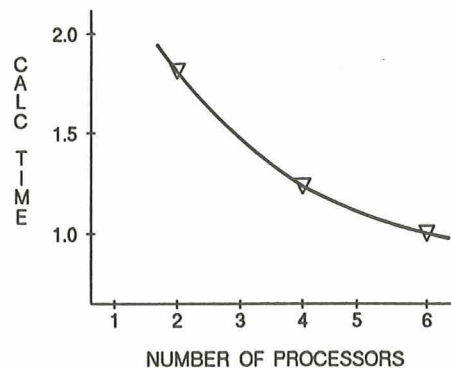
MATRIX INVERSE

$$\begin{bmatrix} A_{1,1} & A_{1,2} & A_{1,3} \\ A_{2,1} & A_{2,2} & A_{2,3} \\ A_{3,1} & A_{3,2} & A_{3,3} \end{bmatrix} * \begin{bmatrix} B_{1,1} & B_{1,2} & B_{1,3} \\ B_{2,1} & B_{2,2} & B_{2,3} \\ B_{3,1} & B_{3,2} & B_{3,3} \end{bmatrix} = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}$$

COMPUTATIONAL TECHNOLOGIES BRANCH

BOUNDARY LAYER ALONG A PLATE

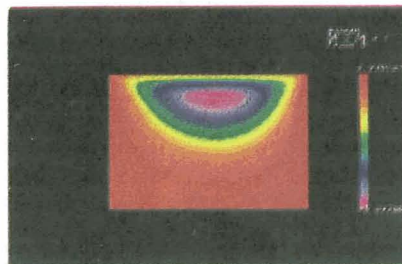
- DISTRIBUTED MEMORY SIMULATION
- SERIAL MATRIX INVERTER



The boundary layer along a flat plate at zero incidence

COMPUTATIONAL TECHNOLOGIES BRANCH

A Parallel, Scalable, 2D Navier-Stokes Solver



Stream function, driven cavity, 50 by 50 grid

- Parallel algorithm design provides efficient implementation on distributed or shared memory machines
 - 74% efficiency achieved on Hypercluster test bed
- Scalable and portable algorithm
 - basic building blocks (e.g. matrix solvers)
 - total solver
- Second-order accurate

COMPUTATIONAL TECHNOLOGIES BRANCH

RELATIONSHIPS

$$\begin{aligned}
 1) \quad & A_{1,1} * B_{1,1} + A_{1,2} * B_{2,1} + A_{1,3} * B_{3,1} = I \\
 2) \quad & A_{2,1} * B_{1,2} + A_{2,2} * B_{2,2} + A_{2,3} * B_{3,2} = I \\
 3) \quad & A_{3,1} * B_{1,3} + A_{3,2} * B_{2,3} + A_{3,3} * B_{3,3} = I \\
 4) \quad & A_{1,1} * B_{1,2} + A_{1,2} * B_{2,2} + A_{1,3} * B_{3,2} = 0 \\
 5) \quad & A_{2,1} * B_{1,3} + A_{2,2} * B_{2,3} + A_{2,3} * B_{3,3} = 0 \\
 6) \quad & A_{3,1} * B_{1,1} + A_{3,2} * B_{2,1} + A_{3,3} * B_{3,1} = 0 \\
 7) \quad & A_{1,1} * B_{1,3} + A_{1,2} * B_{2,3} + A_{1,3} * B_{3,3} = 0 \\
 8) \quad & A_{2,1} * B_{1,1} + A_{2,2} * B_{2,1} + A_{2,3} * B_{3,1} = 0 \\
 9) \quad & A_{3,1} * B_{1,2} + A_{3,2} * B_{2,2} + A_{3,3} * B_{3,2} = 0
 \end{aligned}$$

COMPUTATIONAL TECHNOLOGIES BRANCH

An Efficient and Robust Algorithm For Time Dependent Viscous Incompressible Navier-Stokes Equations

John W. Goodrich
Computational Fluid Dynamics Branch
NASA Lewis Research Center

A recently developed finite difference algorithm is presented for unsteady incompressible Navier-Stokes calculations. The algorithm is extremely robust with respect to Reynolds number, and has been used to directly compute incompressible flows with smoothly resolved streamfunction, kinetic energy and vorticity contours for Reynolds numbers as high as $Re = 100,000$ without requiring any subscale modelling. The accompanying figure shows contour plots for the streamfunction and vorticity at nondimensional times $t = 83$ and $t = 84$ from a transient calculation at $Re = 25,000$, with a 256×256 grid and $\Delta t = \frac{1}{400}$. The algorithm is second order accurate in both time and space, with a Crank-Nicolson differencing for the diffusion terms, with a lagged second order Adams-Basforth differencing for the convection terms, and with central differencing for all space derivatives. There is no constraint on the time step size from diffusion time scales, but the convection time scales impose a stability constraint of $\frac{\delta t}{\delta x} < 1$, and typically a CFL number of 0.75 or 0.80 is used. This algorithm is based on the fourth order Partial Differential Equation for incompressible fluid flow which uses the streamfunction as the only dependent variable,

$$\frac{\partial \Delta \psi}{\partial t} + \frac{\partial \psi}{\partial y} \Delta \frac{\partial \psi}{\partial x} - \frac{\partial \psi}{\partial x} \Delta \frac{\partial \psi}{\partial y} - \frac{1}{Re} \Delta^2 \psi = 0, \quad \text{for } \mathbf{x} \text{ in } \Omega, \text{ and } t > 0.$$

Notice that the vorticity does not enter into this formulation. Although the algorithm currently is only for incompressible flows in two space dimensions, the algorithm can be and is being extended to handle temperature dependant density, subsonic compressible flows, and flows in three space dimensions. The algorithm produces discretely divergence free velocity fields on a nonstaggered grid. The streamfunction discretization is related to a primitive variable discretization on a staggered grid, so that primitive variable velocity boundary conditions can be implemented for the discrete streamfunction. The algorithm is extremely efficient with respect to use of both CPU time and physical memory. A typical time dependant calculation with a 128×128 spatial grid and $\Delta t = \frac{1}{160}$ requires approximately 2 MegaBytes of memory, and approximately 2.3 CPU seconds per time step on an IBM RS/6000 model 530 workstation. Codes implementing the algorithm have been run for problems of various scales on systems ranging from PC's and workstations to CRAY supercomputers. The algorithm has been implemented on an INTEL TOUCHSTONE parallel processor. Solutions will be shown for cavity and channel flows at various Reynolds numbers.

TIME DEPENDENT INCOMPRESSIBLE NAVIER-STOKES EQUATIONS

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u}\mathbf{u}) - \frac{1}{Re} \Delta \mathbf{u} = -\nabla p + \mathbf{F}, \quad \text{for } \mathbf{x} \text{ in } \Omega, \text{ and } t > 0,$$

$$\nabla \cdot \mathbf{u} = 0, \quad \text{for } \mathbf{x} \text{ in } \Omega, \text{ and } t > 0.$$

STREAMFUNCTION EQUATION FOR UNSTEADY INCOMPRESSIBLE FLOW

$$\frac{\partial \Delta \psi}{\partial t} = \frac{1}{Re} \Delta^2 \psi + \frac{\partial \psi}{\partial x} \Delta \frac{\partial \psi}{\partial y} - \frac{\partial \psi}{\partial y} \Delta \frac{\partial \psi}{\partial x}, \quad \text{for } \mathbf{x} \text{ in } \Omega, \text{ and } t > 0;$$

with

$$u(\mathbf{x}, t) = \frac{\partial \psi}{\partial y}, \quad \text{and } v(\mathbf{x}, t) = -\frac{\partial \psi}{\partial x}, \quad \text{for } \mathbf{x} \text{ in } \Omega, \text{ and } t \geq 0.$$

* Initial and boundary conditions must be supplied.

⇒ For $\Omega \subseteq \mathbf{R}^2$.

⇒ Vorticity and pressure **DO NOT** enter into the streamfunction formulation.

⇒ A single equation for a single scalar unknown.

⇒ The velocity solution is always divergence free, and so incompressible.

THE STREAMFUNCTION ALGORITHM FOR UNSTEADY INCOMPRESSIBLE FLOW

$$\begin{aligned} & \text{La}(\tilde{\mathbf{z}}^{n+1}) - \frac{\Delta t}{2Re} \text{Bi}(\tilde{\mathbf{z}}^{n+1}) \\ &= \text{La}(\tilde{\mathbf{z}}^n) + \frac{\Delta t}{2Re} \text{Bi}(\tilde{\mathbf{z}}^n) - \frac{3\Delta t}{2} \left[\delta_x \left(\delta_y(\tilde{\mathbf{z}}^n) \text{La}(\tilde{\mathbf{z}}^n) \right) - \delta_y \left(\delta_x(\tilde{\mathbf{z}}^n) \text{La}(\tilde{\mathbf{z}}^n) \right) \right] \\ & \quad + \frac{\Delta t}{2} \left[\delta_x \left(\delta_y(\tilde{\mathbf{z}}^{n-1}) \text{La}(\tilde{\mathbf{z}}^{n-1}) \right) - \delta_y \left(\delta_x(\tilde{\mathbf{z}}^{n-1}) \text{La}(\tilde{\mathbf{z}}^{n-1}) \right) \right], \end{aligned}$$

with

$$u_{i,j}^n = \frac{1}{2\Delta y} (z_{i,j+1}^n - z_{i,j-1}^n), \quad \text{and } v_{i,j}^n = -\frac{1}{2\Delta x} (z_{i+1,j}^n - z_{i-1,j}^n).$$

* $\tilde{\mathbf{z}}^n \approx \psi(\cdot, t_n)$ on the discrete grid.

* The discretization uses central spatial differencing throughout.

* All variables are defined at each grid point.

* Banded LU decomposition and multigrid solvers have been used.

⇒ The variable ψ is as smooth “as possible”.

⇒ The implicit equation is elliptic, for all Reynolds numbers.

⇒ The local domain of dependance is the large symmetric 13 point discretization stencil.

⇒ The discrete solution is exactly incompressible, $\delta_x(u_{i,j}^n) + \delta_y(v_{i,j}^n) = 0$.

⇒ The stability limit is a CFL constraint, $\frac{\|\mathbf{u}\| \Delta t}{\Delta x} < 1$.

A MULTIGRID SOLVER FOR THE LINEAR IMPLICIT EQUATIONS

$$\text{La}(\tilde{z}^{n+1}) - \frac{\Delta t}{2Re} \text{Bi}(\tilde{z}^{n+1}) = \text{Source Term}(\tilde{z}^n, \tilde{z}^{n-1})$$

- * The Biharmonic operator is factored as two Laplacians.
- * Point Gauss-Seidel smoothing (or Red Black Gauss-Seidel).
- * Linear restriction and prolongation.
- * A V-cycle with 3 iterations per grid level while coarsening, none while refining.

$\Rightarrow \omega = \Delta\psi$ is introduced ONLY for iterating with the biharmonic operator.

\Rightarrow Implemented on scalar, vector and parallel systems.

TYPICAL PERFORMANCE DATA

CAVITY PROBLEM (IBM RS/6000 model 530 workstation)

- * Square cavity transient from $t = 0$ to $t = 1$, at $Re = 9600$.
- * 10 to 15 iteration cycles reduce residuals to less than 5.0×10^{-12} .

- (1) 256×256 grid, $\Delta t = \frac{1}{400}$, 7 grid levels, 6.6 MBytes storage, 10.2 sec/ts;
- (2) 192×192 grid, $\Delta t = \frac{1}{256}$, 7 grid levels, 3.8 MBytes storage, 5.5 sec/ts;
- (3) 128×128 grid, $\Delta t = \frac{1}{160}$, 6 grid levels, 1.7 MBytes storage, 2.5 sec/ts.

\Rightarrow CPU time increases "linearly" with the number of grid points:

- (1) 1.54×10^{-4} sec/ts per grid point on the 256×256 grid;
- (2) 1.48×10^{-4} sec/ts per grid point on the 192×192 grid;
- (3) 1.50×10^{-4} sec/ts per grid point on the 128×128 grid.

\Rightarrow Required memory increases "linearly" with the number of grid points:

- (1) 104.78 Bytes per grid point on the 256×256 grid;
- (2) 106.97 Bytes per grid point on the 192×192 grid;
- (3) 107.12 Bytes per grid point on the 128×128 grid.

\Rightarrow The number of iteration cycles/ts is independent of resolution and Reynolds number.

STREAMFUNCTION MAX AND MIN COMPARATIVE DATA The square driven cavity at Re=5000

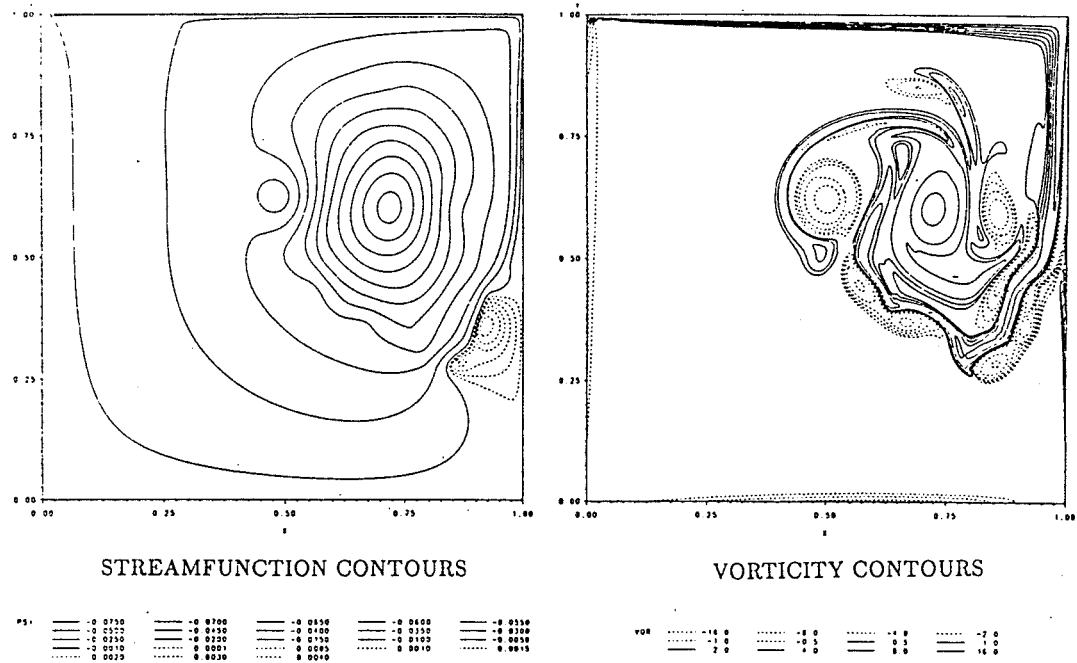
The Streamfunction Minimum

Source	Grid	ψ_{min}	x_{min}	y_{min}
Kim and Moin	96×96	-1.12×10^{-1}		
Goodrich	128×128	-1.15×10^{-1}	$\frac{66}{128} = \frac{132}{256}$	$\frac{69}{128} = \frac{138}{256}$
Goodrich	256×256	-1.18×10^{-1}	$\frac{132}{256}$	$\frac{137}{256}$
Ghia, Ghia, and Shin	256×256	-1.19×10^{-1}	$\frac{131}{256}$	$\frac{137}{256}$

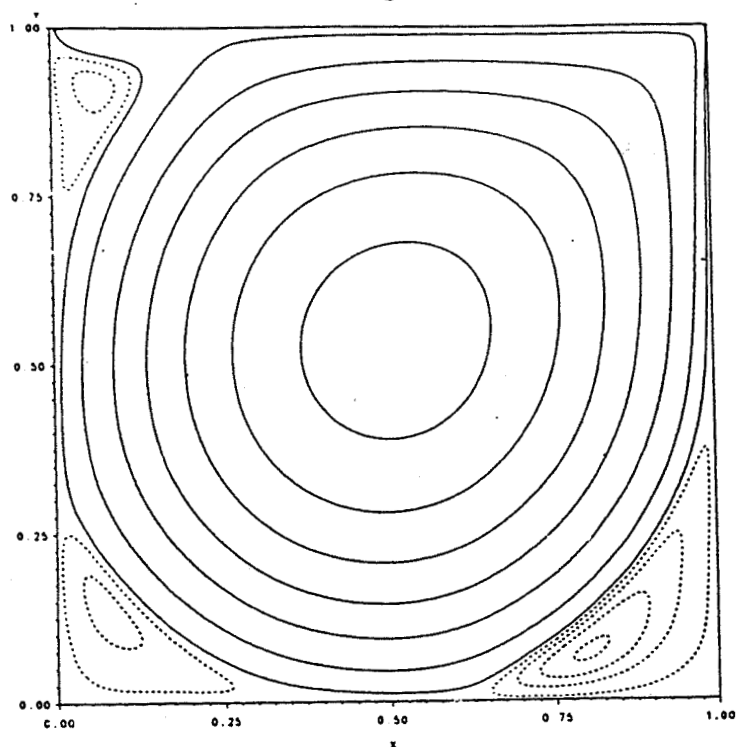
The Streamfunction Maximum

		ψ_{max}	x_{max}	y_{max}
Goodrich	128×128	3.44×10^{-3}	$\frac{102}{128} = \frac{204}{256}$	$\frac{9}{128} = \frac{18}{256}$
Goodrich	256×256	3.13×10^{-3}	$\frac{206}{256}$	$\frac{19}{256}$
Ghia, Ghia, and Shin	256×256	3.08×10^{-3}	$\frac{207}{256}$	$\frac{19}{256}$

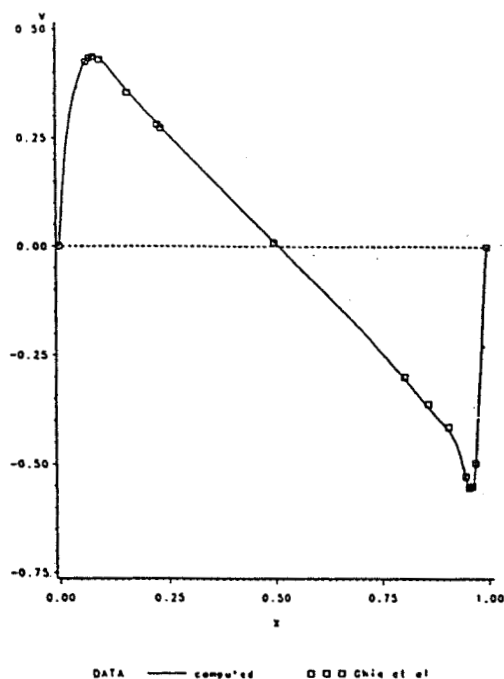
TRANSIENT DRIVEN CAVITY $Re = 25,000$, 256×256 grid, $t = 10.25$



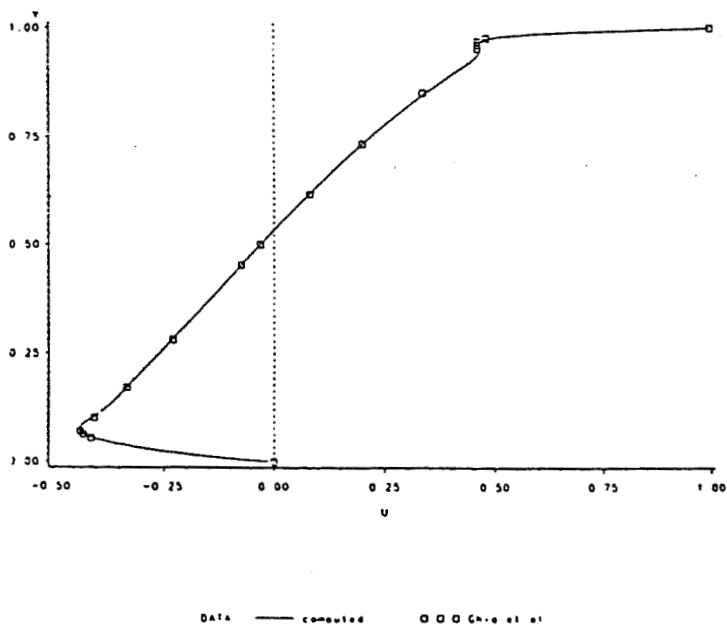
STREAM FUNCTION CONTOURS
Re=5k, 256*256 grid, t=748.45



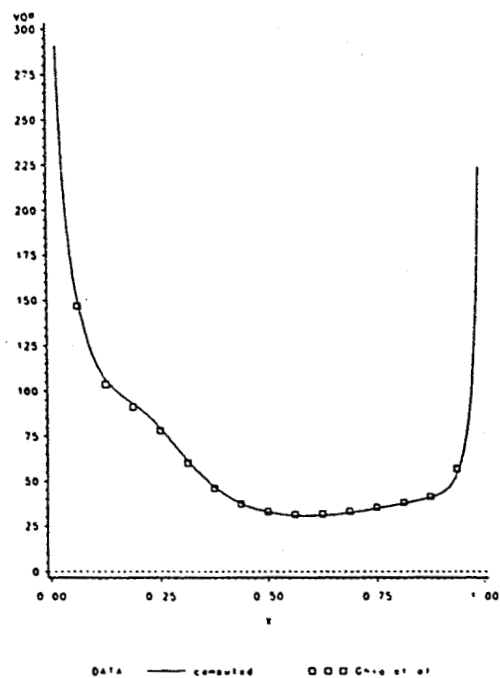
Re=5000, 256 by 256 grid, t=748.45
v at y=0.5 as a function of x



Re=5000, 256 by 256 grid, t=748.45
u at x=0.5 as a function of y



Re=5000, 256 by 256 grid, t=748.45
vorticity at y=1.0 as a function of x



Re = 5000, 128 × 128 grid, $\Delta t = \frac{1}{160}$, $t = 748.45$, calculated data,
compared with U. Ghia, K. N. Ghia and C. T. Shin, *J. Comput. Phys.* 48 , 387 (1982).

TYPICAL PERFORMANCE DATA

CHANNEL PROBLEM (IBM RS/6000 model 530 workstation)

- * Backward facing half height step with a parabolic inflow profile on the step corner.
- * Transient from $t = 0$ to $t = 100$, $Re = 800$, $u_{max} = 1.5$ for inflow profile, $\Delta t = \frac{1}{100}$.
- * Outflow Boundary Conditions are:

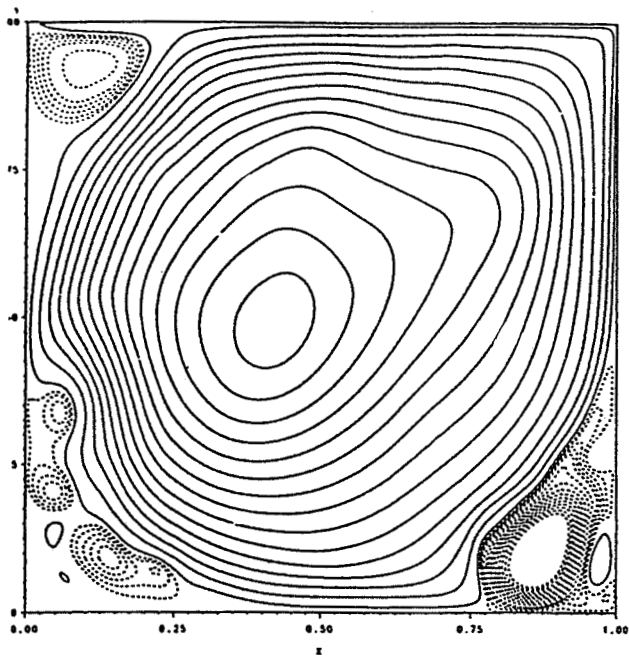
$$\frac{\partial^2 \psi}{\partial x^2} = 0 \text{ and } \frac{\partial^2 \omega}{\partial x^2} = 0 \Leftrightarrow \frac{\partial^2 \psi}{\partial x^2} = 0 \text{ and } \frac{\partial^4 \psi}{\partial x^4} = 0$$

- * 10 to 35 iteration cycles reduce residuals to less than 5.0×10^{-12}
 - * 30 iterations per cycle on the coarse grid.
 - * Coded for 1024×32 grid with 4 grid levels using 3.5 MBytes storage.
- (1) 800×32 grid, 25 diameter domain, 4.7 s/ts;
 - (2) 480×32 grid, 15 diameter domain, 3.1 s/ts;
 - (3) 224×32 grid, 7 diameter domain, 1.7 s/ts.

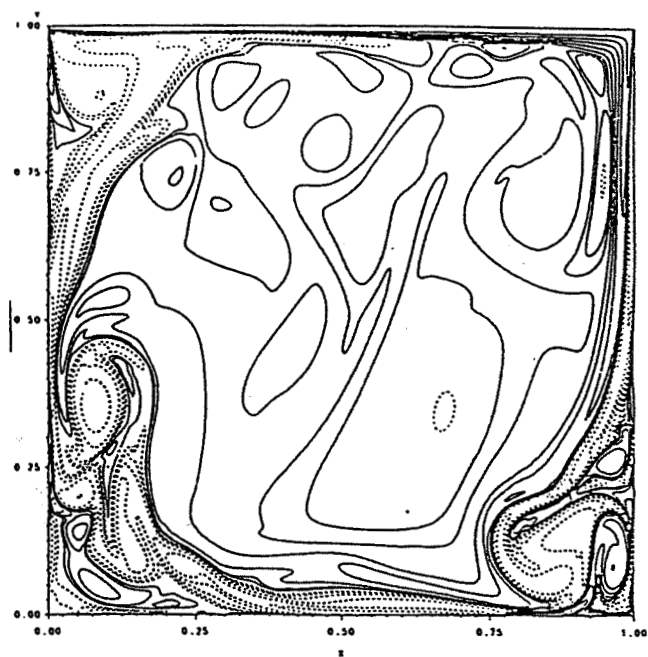
⇒ Relative CPU time decreases with the length of the domain:

- (1) 1.79×10^{-4} sec/ts per grid point on the 800×32 grid;
- (2) 1.95×10^{-4} sec/ts per grid point on the 480×32 grid;
- (3) 2.23×10^{-4} sec/ts per grid point on the 224×32 grid.

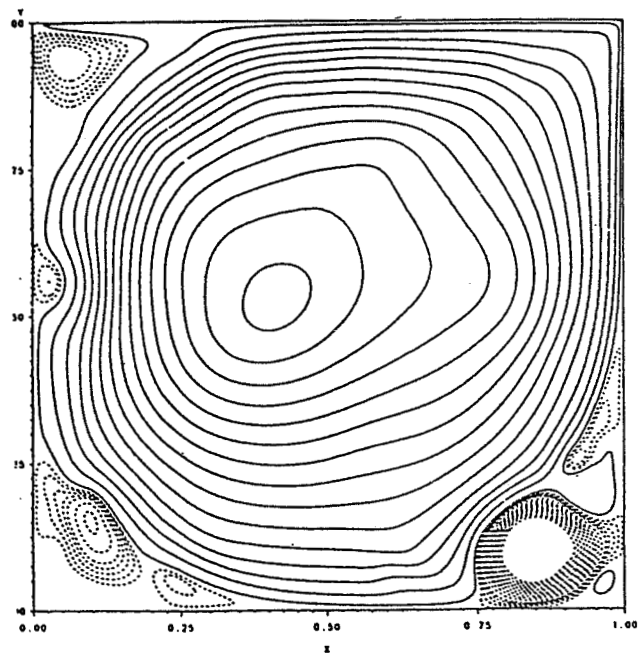
STREAM FUNCTION CONTOURS
Re=25000, 256*256 grid, t=83.00



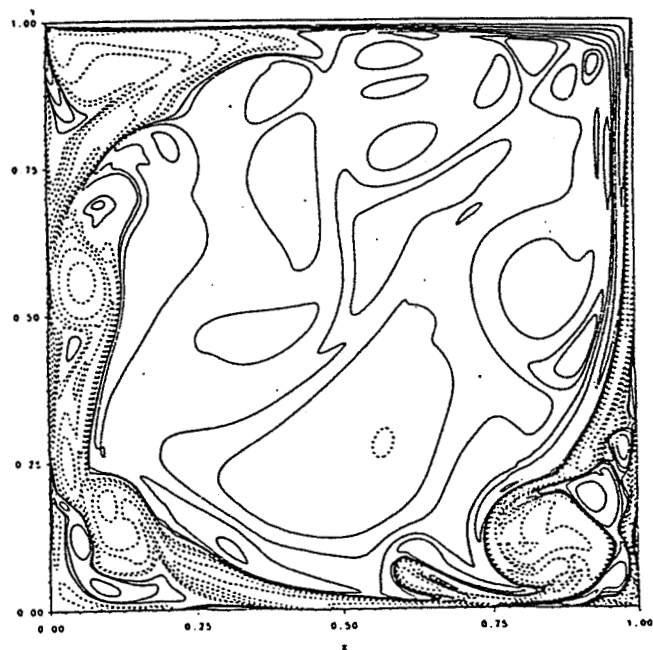
VORTICITY CONTOURS
Re=25000, 256*256 grid, t=83.00



STREAM FUNCTION CONTOURS
Re=25000, 256*256 grid, t=84.00



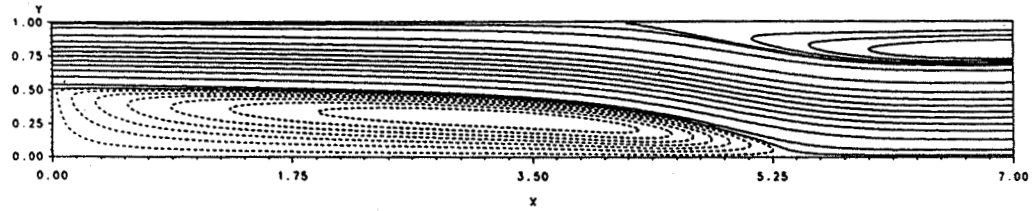
VORTICITY CONTOURS
Re=25000, 256*256 grid, t=84.00



----- -0.0750	----- -0.0700	----- -0.0650	----- -0.0600	----- -0.0550
----- -0.0500	----- -0.0450	----- -0.0400	----- -0.0350	----- -0.0300
----- -0.0250	----- -0.0200	----- -0.0150	----- -0.0100	----- -0.0050
----- -0.0010	----- 0.0001	----- 0.0005	----- 0.0010	----- 0.0015
----- 0.0020	----- 0.0030	----- 0.0040	----- 0.0050	----- 0.0060
----- 0.0070	----- 0.0080	----- 0.0090	----- 0.0100	----- 0.0110
----- 0.0120				

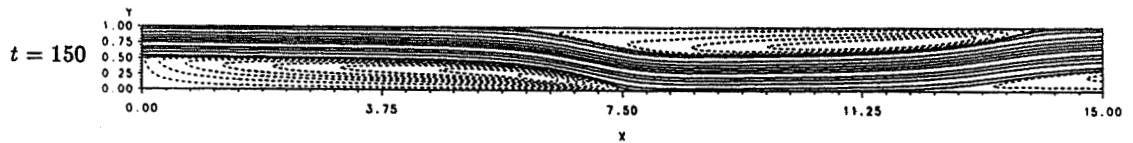
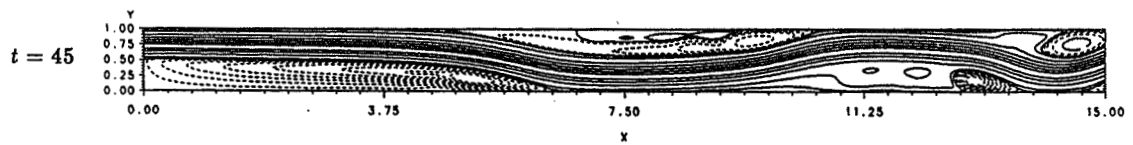
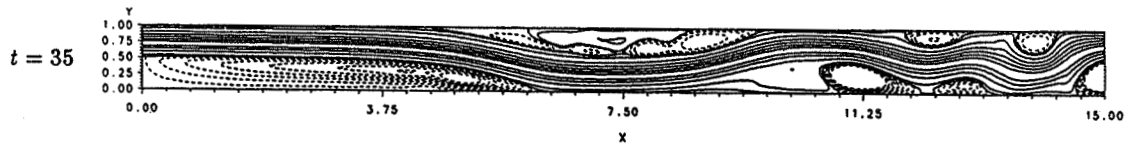
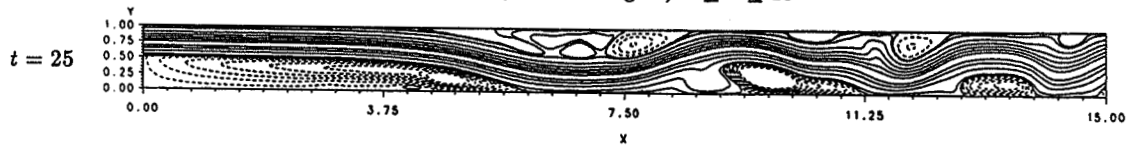
----- -32.0	----- -16.0	----- -8.0	----- -4.0	----- -2.0
----- -1.0	----- 0.0	----- 8.0	----- 16.0	----- 32.0

STREAM FUNCTION CONTOURS
 $Re=800$, 64×448 grid, $0 \leq x \leq 7$, $t=30$, $u_l=1.5$
 Backward facing step with parabolic inflow, code g4a32vc3.f



PS1	----- -0.030	----- -0.025	----- -0.020	----- -0.015	----- -0.010	----- -0.005
	----- -0.001	===== 0.001	===== 0.010	===== 0.050	===== 0.100	===== 0.150
	===== 0.200	===== 0.250	===== 0.300	===== 0.350	===== 0.400	===== 0.450
	===== 0.490	===== 0.499	===== 0.500	===== 0.502	===== 0.504	===== 0.508

BACKWARD FACING STEP
 $Re = 2000$, 32×480 grid, $0 \leq x \leq 15$



STREAMFUNCTION CONTOURS

PS1	----- -0.035	----- -0.030	----- -0.025	----- -0.020	----- -0.015	----- -0.010	----- -0.005
	----- -0.001	===== 0.001	===== 0.010	===== 0.050	===== 0.100	===== 0.150	===== 0.100
	===== 0.250	===== 0.300	===== 0.350	===== 0.400	===== 0.450	===== 0.490	===== 0.499
	===== 0.500	===== 0.525	===== 0.550	===== 0.575	===== 0.600		

SUMMARY

- * Vorticity and pressure **DO NOT** enter into the streamfunction formulation.
 - * A single equation for a single scalar unknown in \mathbb{R}^2 .
 - * The variable ψ is as smooth "as possible".
-
- \Rightarrow Second order accuracy in both time and space.
 - \Rightarrow The discrete solution is exactly incompressible, $\delta_x(u_{i,j}^m) + \delta_y(v_{i,j}^m) = 0$.
 - \Rightarrow The stability limit is a CFL constraint, $\frac{\|u\|\Delta t}{\Delta x} < 1$.
 - \Rightarrow Implemented on scalar, vector and parallel systems.
 - \Rightarrow CPU time increases "linearly" with the number of grid points.
 - \Rightarrow Required memory increases "linearly" with the number of grid points.
 - \Rightarrow The number of iteration cycles/ts is independent of resolution and Reynolds number.
 - \Rightarrow Extremely robust with respect to Reynolds number.

GOALS

- * Develop and adapt numerical methods.
- * Investigate dynamic phenomena (instable, transitional, unsteady time asymptotic).

STEPS

- * Incompressible flow (robust, fast, validated).
- * Artificial outflow boundary conditions (Tom Hagstrom, Univ. of N.M.).
- * Parallel processor implementation on INTEL iPSC (Rodger Dyson, NASA Lewis).
- * Temperature dependant density (Boussinesq approximation).
- * Chemistry models (Marty Rabinowitz, NASA Lewis).
- * Multilevel adaptive methods (Steve McCormick, Univ. Co. Denver).
- * Compressible subsonic flow.
- * Higher order differencing scheme.
- * Three dimensional version.

APPLICATIONS

- * Development of models for simplified chemical kinetics.
- * Development of models for chemistry/turbulence interaction.
- * Prediction of chemical emissions (High Speed Research Program).
- * Understanding instabilities (flashback, flameout and accoustic/combustion interaction).

ADVANCES IN ENGINEERING TURBULENCE MODELING

T.-H. Shih

Institute for Computational Mechanics in Propulsion
NASA Lewis Research Center

Some new developments in two-equation models and second order closure models will be presented. Two-equation models (e.g., k - ϵ model) have been widely used in CFD for engineering problems. In most low-Reynolds number two-equation models, some wall-distance damping functions are used, especially in the eddy viscosity, to account for the effect of wall on turbulence. However, this often causes the confusions and difficulties in computing flows with complex geometry and also needs an ad hoc treatment near the separation and reattachment points. In this paper, modified two-equation models are proposed to remove abovementioned shortcomings. The calculations using various two-equation models are compared with direct numerical simulations of channel flows and flat boundary layers.

Development of second order closure model will be also discussed with emphasis on the modeling of pressure related correlation terms and dissipation rates in the second moment equations. All the existing models poorly predict the normal stresses near the wall and fail to predict the 3 dimensional effect of mean flow on the turbulence (e.g., decrease in the shear stress caused by the cross flow in the boundary layer). The newly developed second order near-wall turbulence model to be described in this paper is capable of capturing the near-wall behavior of turbulence as well as the effect of three dimension mean flow on the turbulence.

- Two-Equation Near-Wall Turbulence Models
- Second-Order Closure Models
- Modeling of Three-Dimensional Turbulent Flows

k - ϵ model

The eddy viscosity ν_T is assumed in two-equation models as follows:

$$\nu_T = C_\mu f_\mu \frac{k^2}{\epsilon}$$

$$\nu_T = C_\mu f_\mu k \tau$$

where $\tau = k/\epsilon$.

The general k - ϵ (or k - τ) model equations are of the following forms:

$$k_{,t} + U_j k_{,j} = \left[\left(\frac{\nu_T}{\sigma_k} + \nu \right) k_{,j} \right]_{,j} + \Pi + \nu_T (U_{i,j} + U_{j,i}) U_{i,j} - \epsilon + D$$

$$\epsilon_{,t} + U_j \epsilon_{,j} = \left[\left(\frac{\nu_T}{\sigma_\epsilon} + \nu \right) \epsilon_{,j} \right]_{,j} + C_1 f_1 \frac{\epsilon}{k} \nu_T (U_{i,j} + U_{j,i}) U_{i,j} - C_2 f_2 \frac{\epsilon \tilde{\epsilon}}{k} + E$$

$$\tau_{,t} + U_j \tau_{,j} = \left[\left(\frac{\nu_T}{\sigma_{\tau 2}} + \nu \right) \tau_{,j} \right]_{,j} + \frac{2}{k} \left(\frac{\nu_T}{\sigma_{\tau 1}} + \nu \right) k_{,i} \tau_{,i} - \frac{2}{\tau} \left(\frac{\nu_T}{\sigma_{\tau 1}} + \nu \right) \tau_{,i} \tau_{,i}$$

$$+ (1 - C_{\epsilon 1}) \frac{\tau}{k} \nu_T (U_{i,j} + U_{j,i}) U_{i,j} + (C_{\epsilon 2} f_2 - 1)$$

The damping function f_μ :

$$f_\mu = f\left(\frac{k^2}{\nu\epsilon}\right) = f(R_t) \text{ — Jones and Launder model}$$

$$f_\mu = f\left(\frac{u_\tau y}{\nu}\right) = f(y^+) \text{ — for all other } k\text{-}\epsilon \text{ models}$$

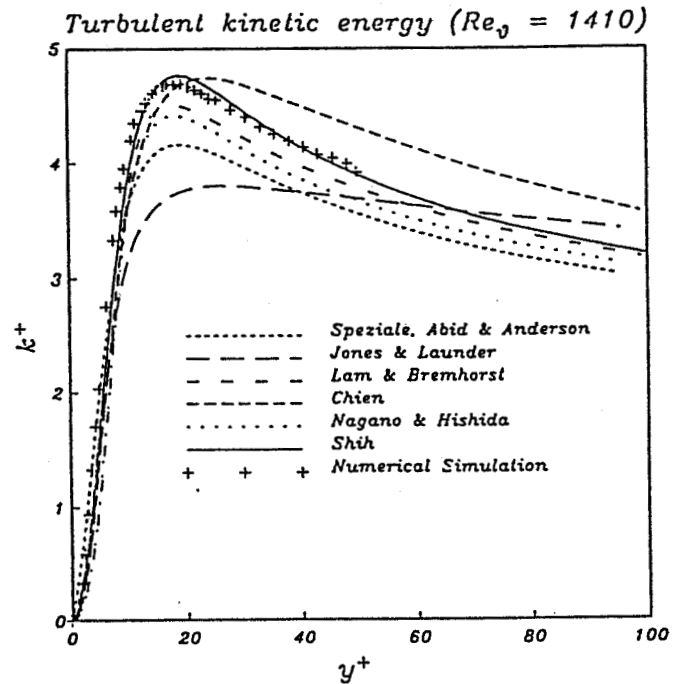
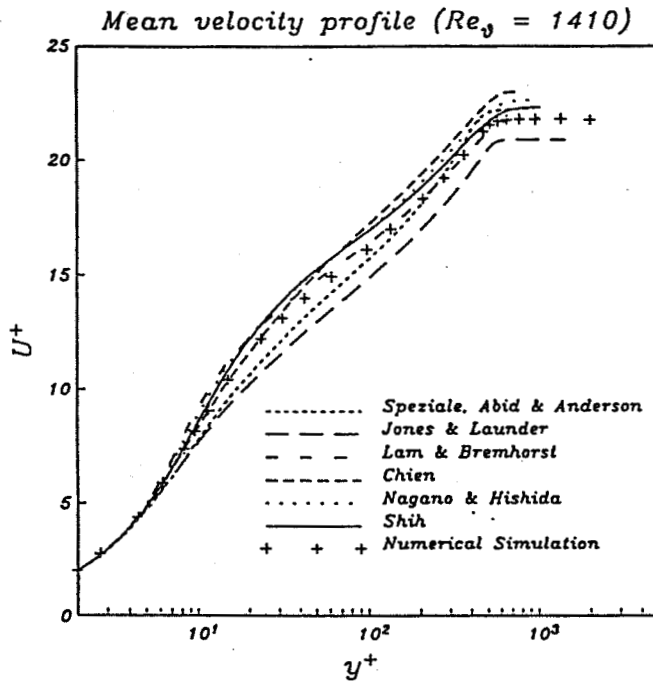
The parameter R_t and y^+ are used for counting the wall effect.

The problems with the form of $f_\mu(y^+)$ are following:

Near the separation region, $f_\mu \rightarrow 0$, $\Rightarrow \nu_T = 0$;

In the flows with complex geometry, the y is not well defined.

y^+ is an unacceptable parameter.



In the present study, we suggest another parameter:

$$R = \frac{k^{3/2}}{\epsilon} \frac{|U|}{\nu}$$

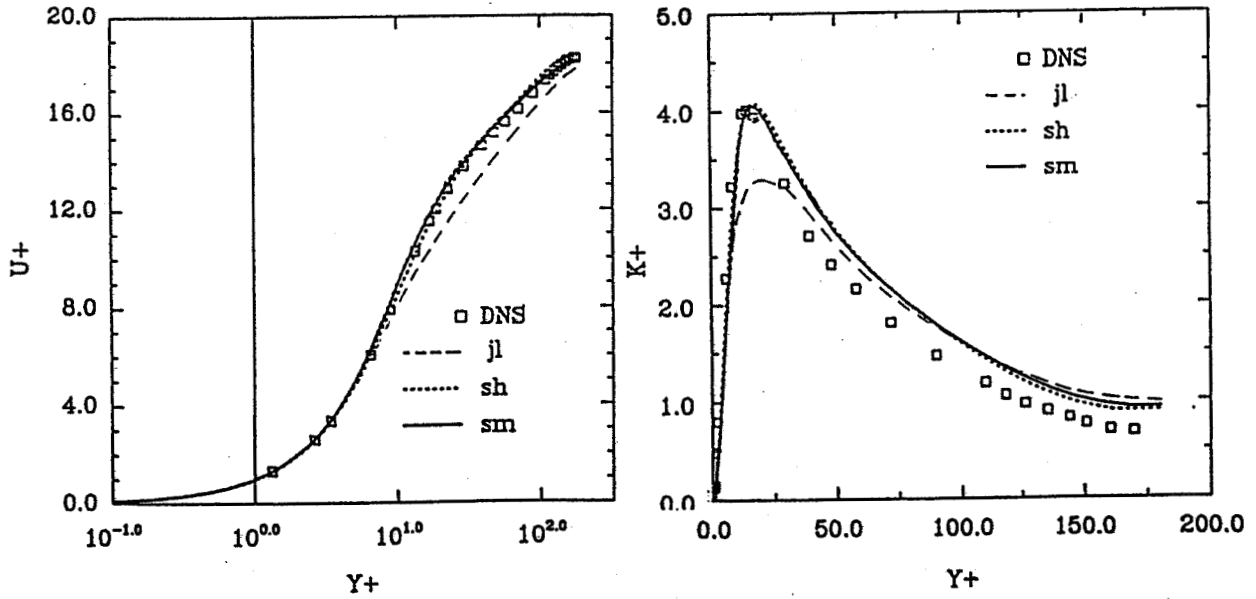
and the following modifications in Shih's model:

$$f_\mu = 1 - \exp \left\{ C_3 \left[1 - \exp(C_6 R^{1/4}) \right] \right\}$$

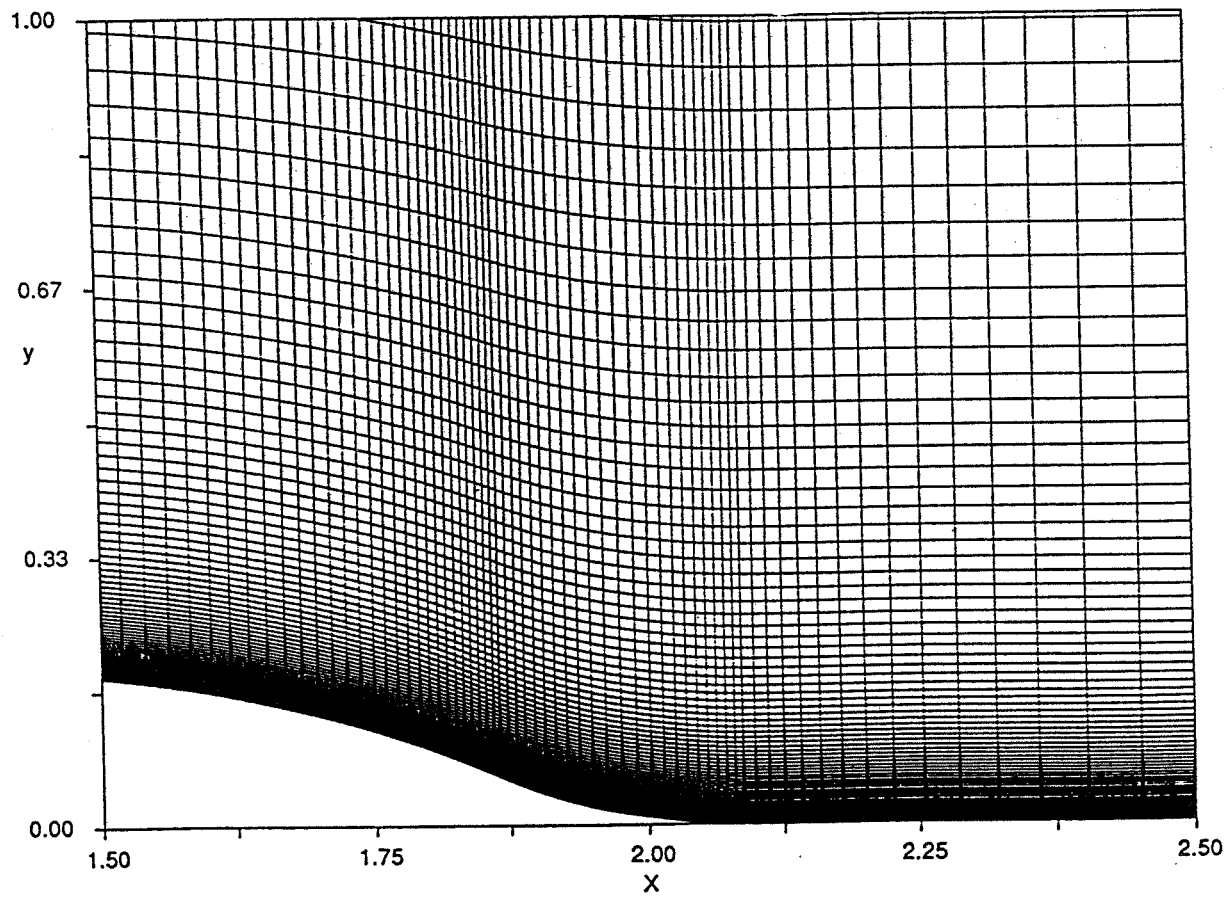
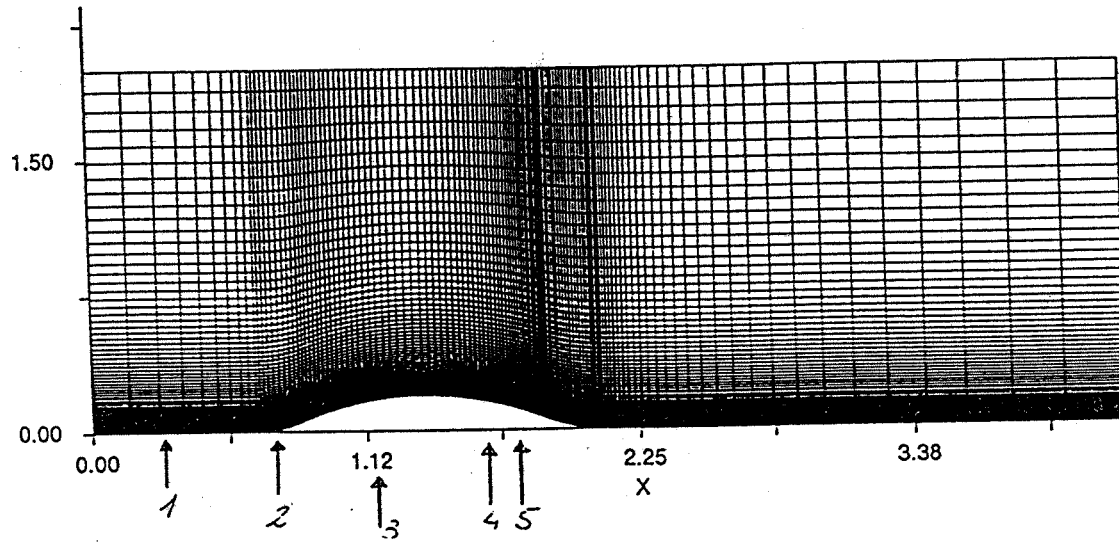
$$\Pi = \left[\frac{C_0 \nu_T}{f_\mu^2 \sigma_k} k_{,j} \right]_{,j}$$

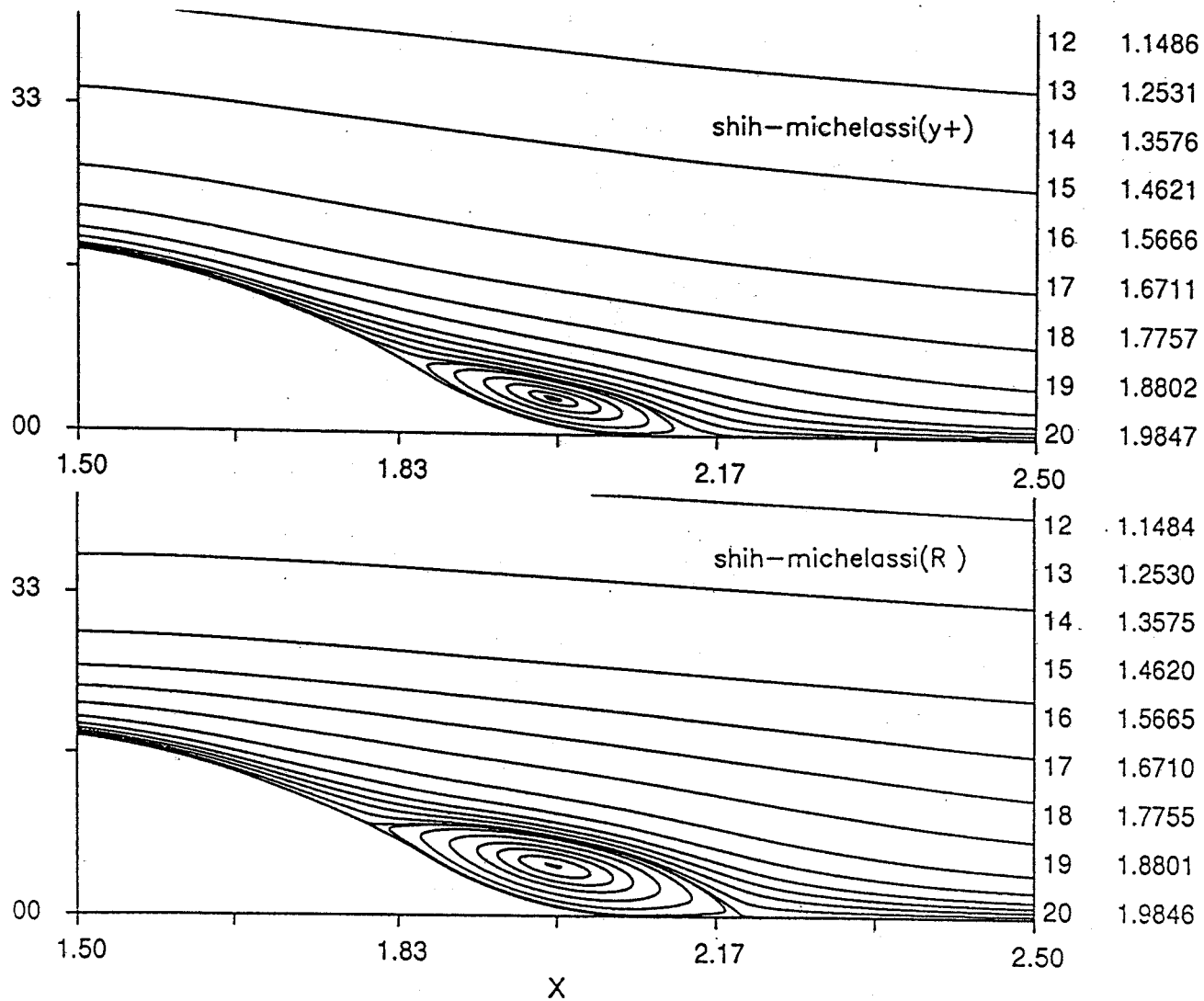
$$\tilde{\epsilon} = \epsilon \left[1 - \exp(-R_t^{1/2}) \right]$$

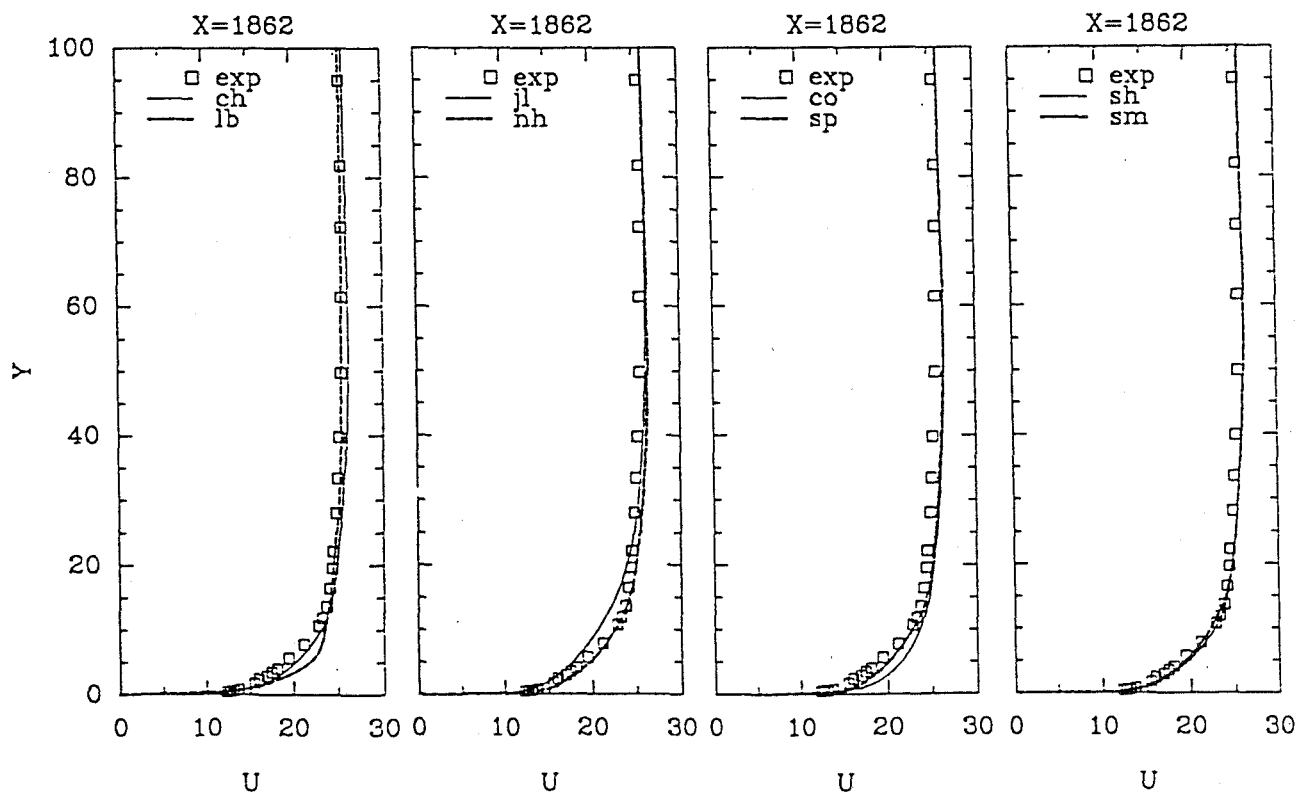
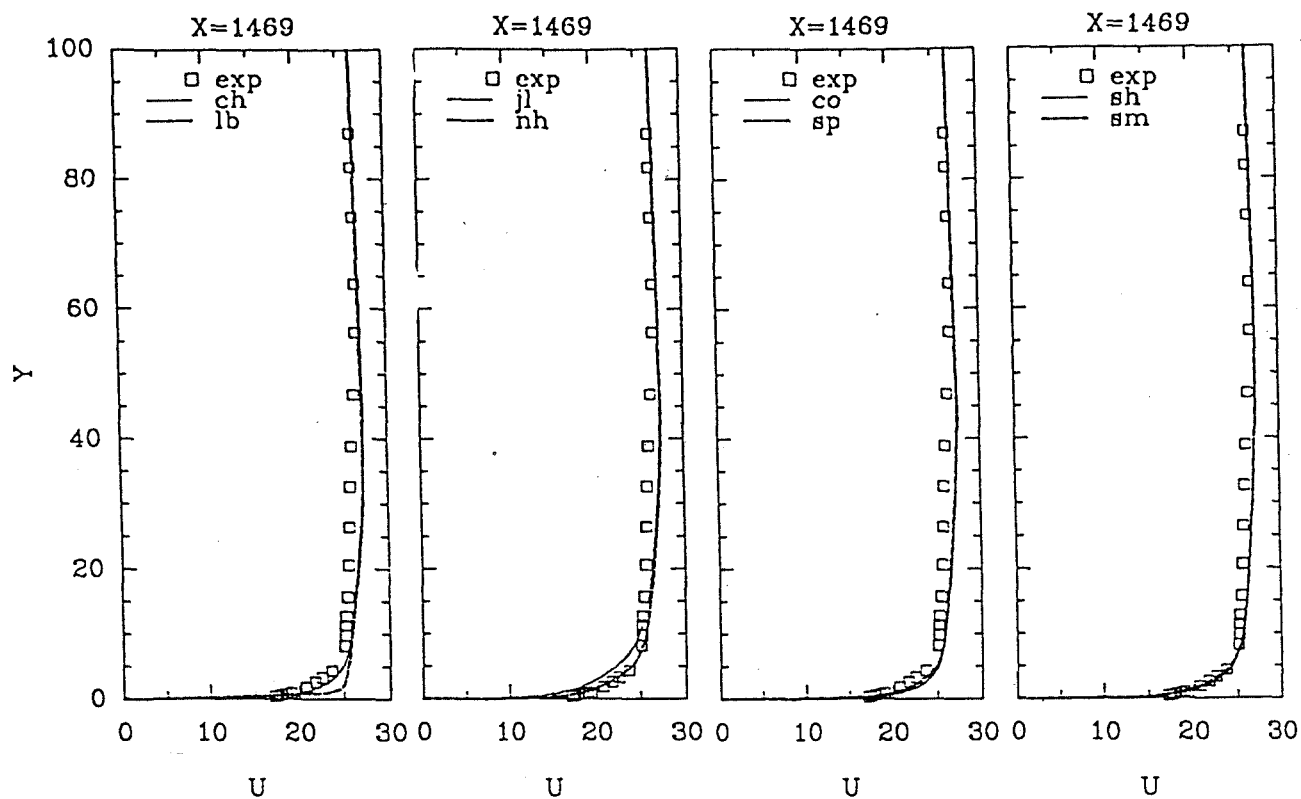
where $R_t = k^2 / \nu \epsilon$, $C_0 = .004$, $C_3 = .0004$, $C_6 = 1.2$

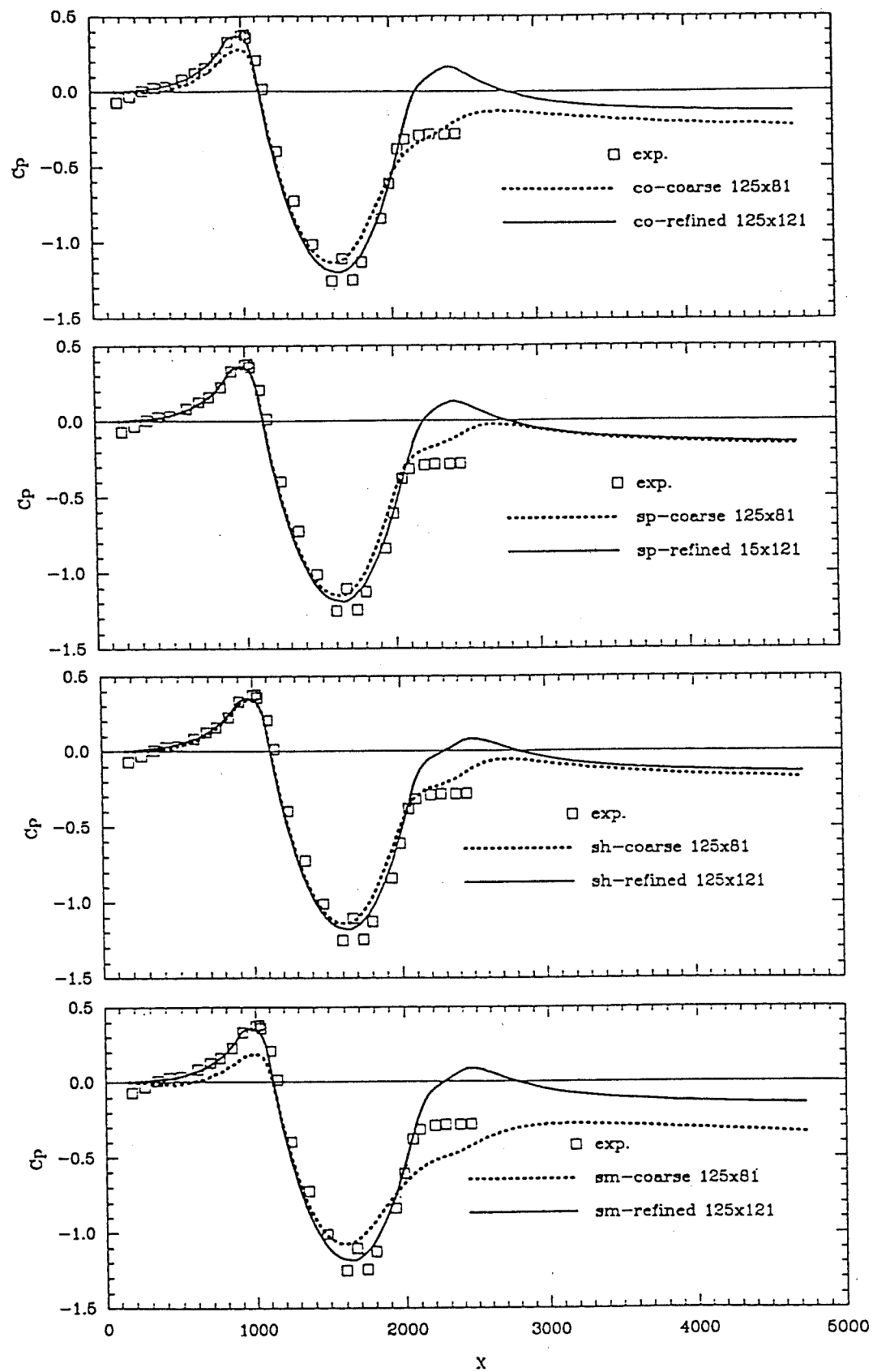


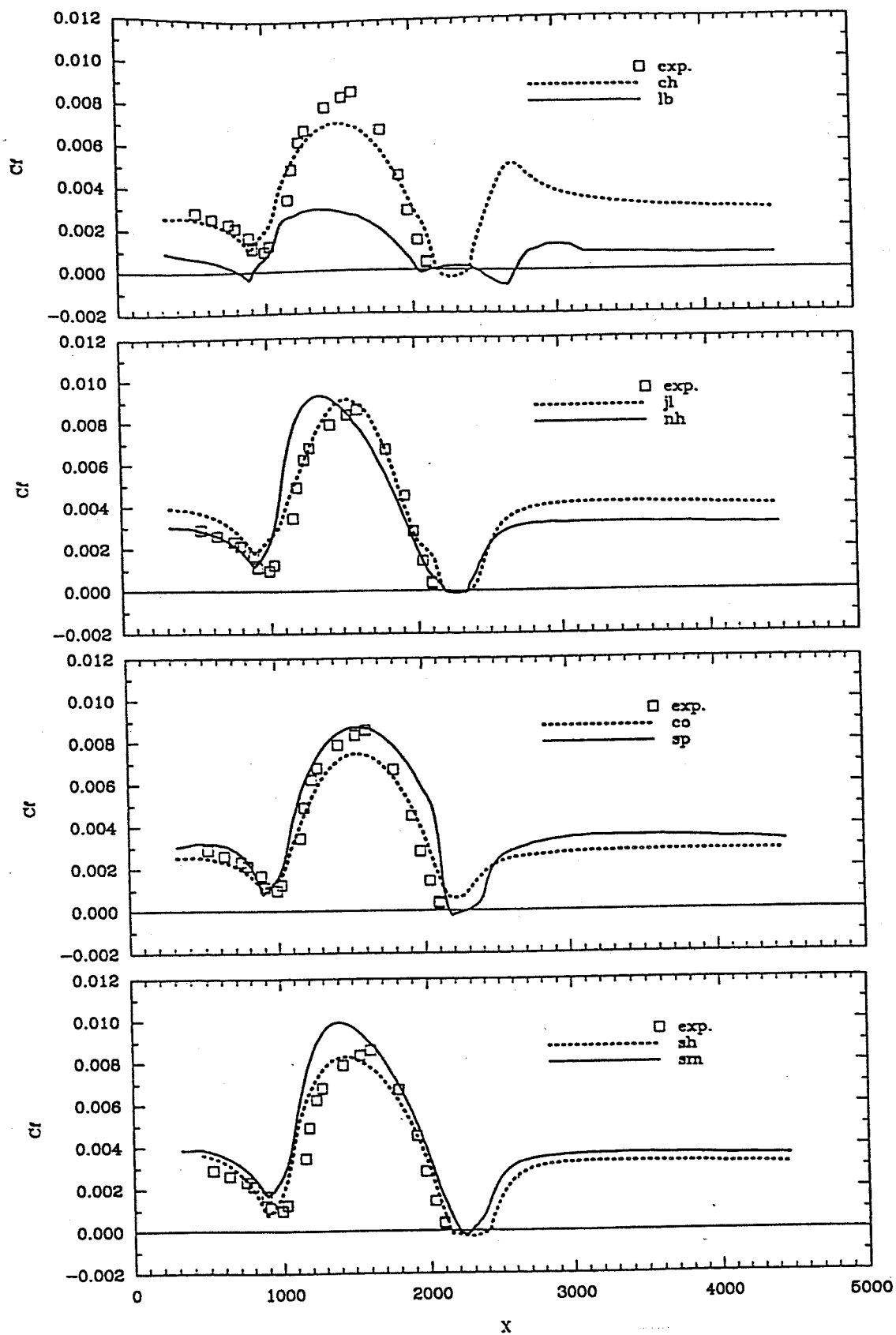
Turbulent flow past a hill ($Re \approx 1.4 \cdot 10^6$)











Another alternative:

$$R_u = \frac{U^4}{\nu \epsilon}$$

and

$$f_\mu = 1 - \exp(-a_1 R_u^{1/4} - a_2 R_u^{1/2} - a_3 R_u)$$

$$\Pi = \left[\frac{C_0}{f_\mu^2} \frac{\nu_T}{\sigma_k} k_{,j} \right]_{,j}$$

$$\tilde{\epsilon} = \epsilon \left[1 - \exp(-R_t^{1/2}) \right]$$

where $R_t = k^2/\nu\epsilon$, $a_1 = 5 \cdot 10^{-3}$, $a_2 = \cdot 10^{-5}$, $a_3 = 8 \cdot 10^{-7}$, $C_0 = .01$.

Second order modeling of near-wall turbulence

The exact equation for the Reynolds stress tensor is:

$$\frac{D}{Dt} \langle u_i u_j \rangle = P_{ij} + T_{ij} + D_{ij}^{(\nu)} + \Pi_{ij} - \epsilon_{ij}$$

where $\langle \rangle$ stands for an ensemble average, $D/Dt = \partial/\partial t + U_k \partial/\partial x_k$.

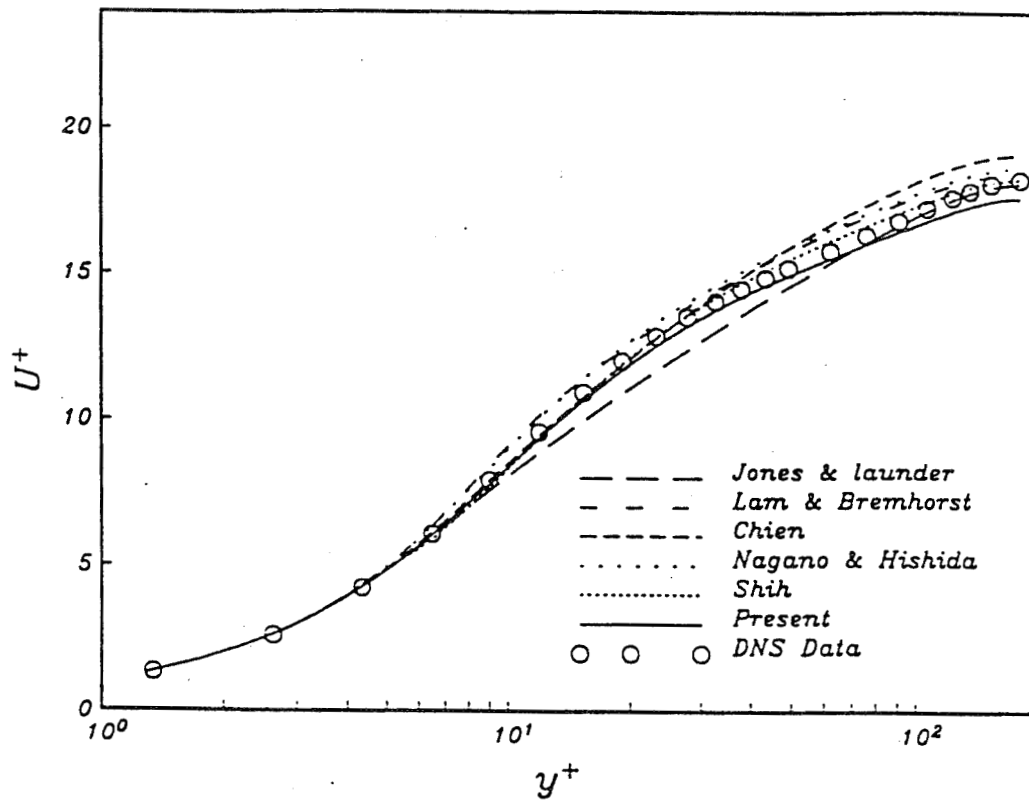
$$P_{ij} = -\langle u_i u_k \rangle U_{j,k} - \langle u_j u_k \rangle U_{i,k}$$

$$T_{ij} = -\langle u_i u_j u_k \rangle_{,k}$$

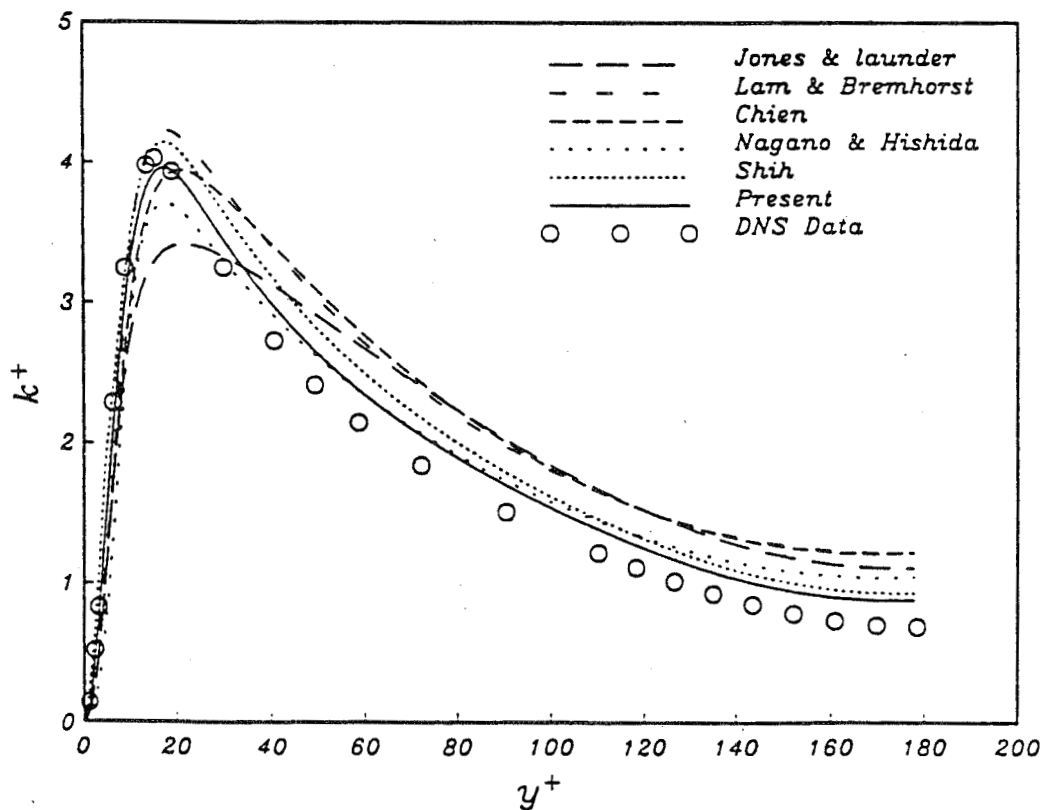
$$D_{ij}^{(\nu)} = \nu \langle u_i u_j \rangle_{,kk}$$

$$\Pi_{ij} = -\frac{1}{\rho} \langle u_i p_{,j} + u_j p_{,i} \rangle$$

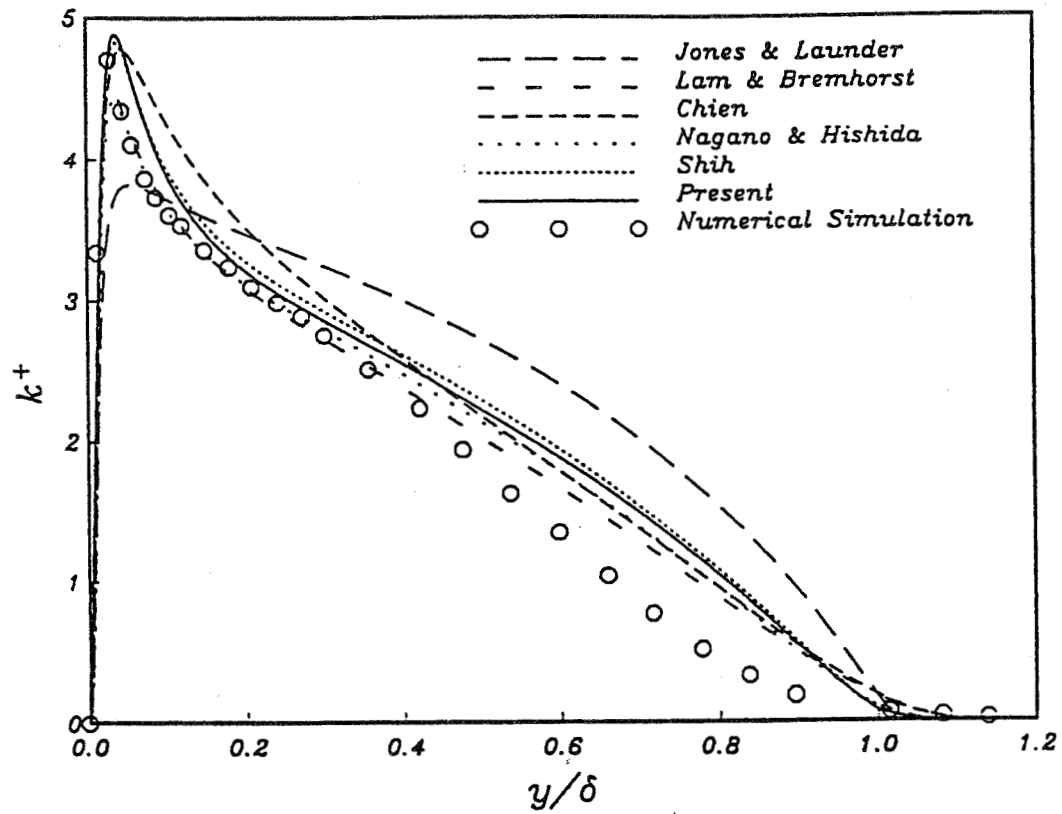
$$\epsilon_{ij} = 2\nu \langle u_{i,k} u_{j,k} \rangle$$



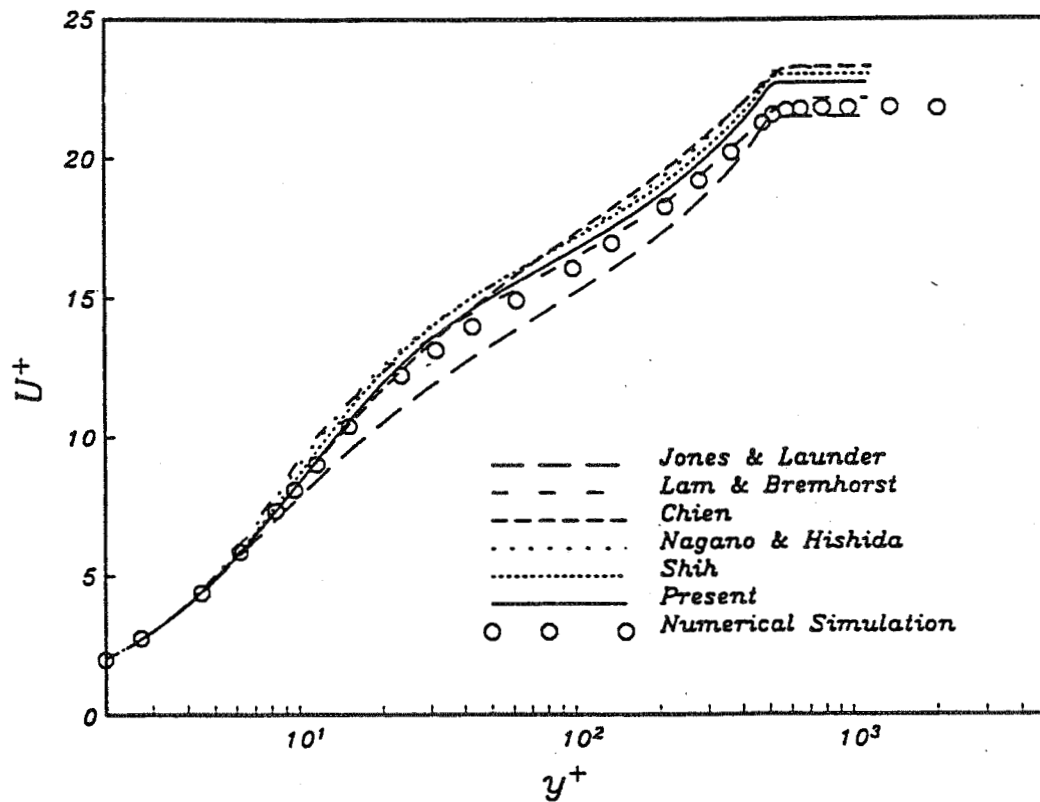
Mean velocity profile for channel flow, $Re_\tau = 180$.



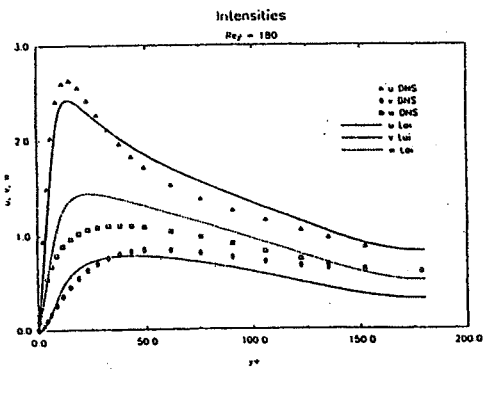
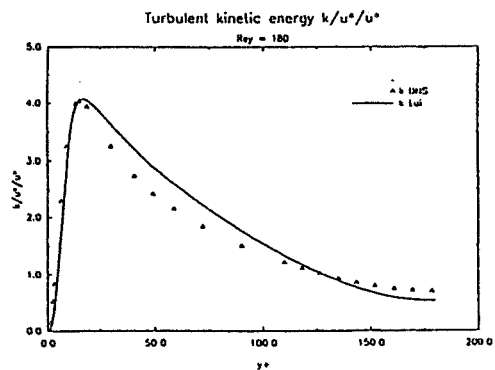
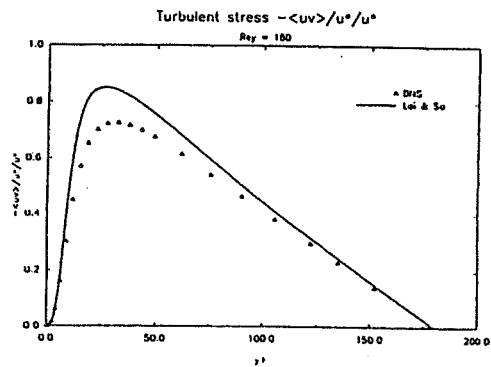
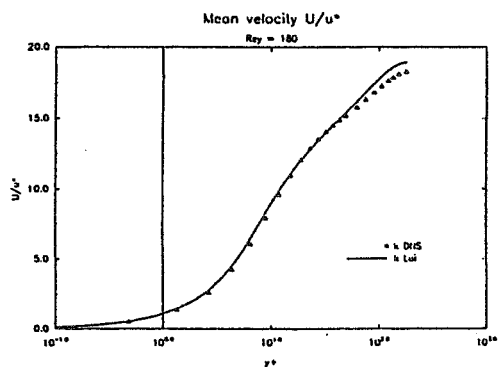
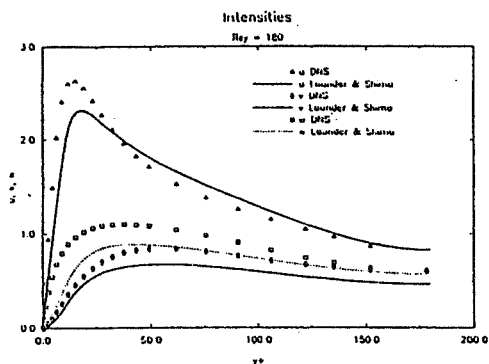
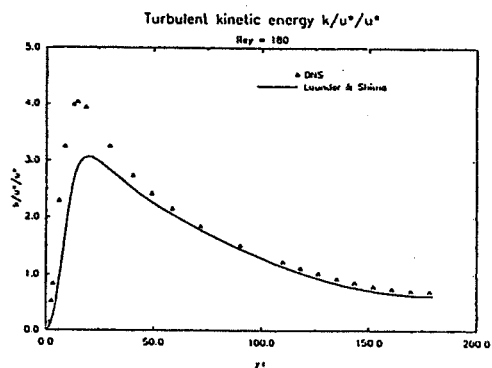
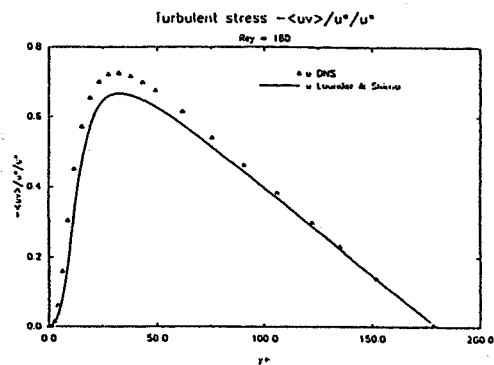
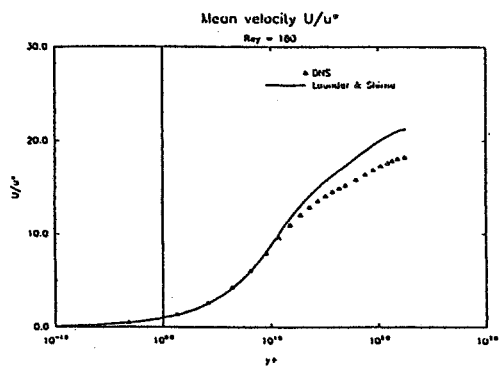
Turbulent kinetic energy for channel flow, $Re_\tau = 180$.



Turbulent kinetic energy for flat plate boundary layer, $Re_\theta = 1410$.



Mean velocity profile for flat plate boundary layer, $Re_\theta = 1410$.



Using the near-wall asymptotic behavior of turbulence^[10] as model constraints, we formed a set of modeled transport equations for the Reynolds-stress tensor and the dissipation rate of turbulent kinetic energy.

The proposed near-wall model for $\Pi_{ij} - \varepsilon_{ij}$ is:

$$\Pi_{ij} - \varepsilon_{ij} = -f_w \frac{\epsilon}{\langle q^2 \rangle} [2\langle u_i u_j \rangle + 4(\langle u_i u_k \rangle n_j n_k + \langle u_j u_k \rangle n_i n_k) + 2\langle u_k u_l \rangle n_k n_l n_i n_j]$$

where n_i is a unit vector normal to the surface, and $f_w = \exp(-(R_t/C_1)^2)$, $R_t = \frac{\langle q^2 \rangle^2}{9\nu\epsilon}$, $C_1 = 1.358R_{e\tau}^{0.44}$, $R_{e\tau} = u_\tau \delta/\nu$. u_τ is the friction velocity, δ is the thickness of the boundary layer or the half width of the channel.

Away from the wall, the velocity pressure-gradient correlation Π_{ij} is split into the rapid part $\Pi_{ij}^{(1)}$ and the slow part $\Pi_{ij}^{(2)}$:

$$\Pi_{ij} = \Pi_{ij}^{(1)} + \Pi_{ij}^{(2)}$$

The proposed model for $\Pi_{ij}^{(2)} - \varepsilon_{ij}$ is:

$$\Pi_{ij}^{(2)} - \varepsilon_{ij} = -\epsilon(\beta b_{ij} + \frac{2}{3}\delta_{ij})(1 - f_w)$$

where

$$\beta = 2 + \frac{F}{9} \left\{ \frac{72}{R_t^{1/2}} + 80.1 \ln[1 + 62.4(-II + 2.3III)] \right\} \exp\left(-\frac{7.77}{R_t^{1/2}}\right)$$

$$F = 1 + 27III + 9II$$

$$II = -\frac{1}{2}b_{ij}b_{ji}$$

$$III = \frac{1}{3}b_{ij}b_{jk}b_{ki}$$

$$b_{ij} = \langle u_i u_j \rangle / \langle q^2 \rangle - \delta_{ij}/3$$

The rapid part of velocity pressure-gradient, $\Pi_{ij}^{(1)}$ is modeled as follows(Shih and Lumley^[11,12]):

$$\begin{aligned}\Pi_{ij}^{(1)} = & \left(\frac{1}{5} + 2a_5\right)\langle q^2 \rangle (U_{i,j} + U_{j,i}) - \frac{2}{3}(1 - a_5)(P_{ij} - \frac{2}{3}P\delta_{ij}) \\ & + \left(\frac{2}{3} + \frac{16}{3}a_5\right)(D_{ij} - \frac{2}{3}P\delta_{ij}) + \frac{2}{15}(P_{ij} - D_{ij}) + \frac{6}{5}b_{ij}P \\ & + \frac{2}{5\langle q^2 \rangle} [\langle u_i u_k \rangle U_{j,q} + \langle u_j u_k \rangle U_{i,q} \langle u_k u_q \rangle - \langle u_i u_p \rangle \langle u_j u_q \rangle (U_{p,q} + U_{q,p})]\end{aligned}$$

where,

$$\begin{aligned}P_{ij} &= -\langle u_i u_k \rangle U_{j,k} - \langle u_j u_k \rangle U_{i,k} \\ D_{ij} &= -\langle u_i u_k \rangle U_{k,j} - \langle u_j u_k \rangle U_{k,i} \\ P &= \frac{1}{2}P_{ii} \\ a_5 &= -\frac{1}{10}(1 + C_2 F^{1/2}) \\ C_2 &= 0.8[1 - \exp(-(R_t/40)^2)]\end{aligned}$$

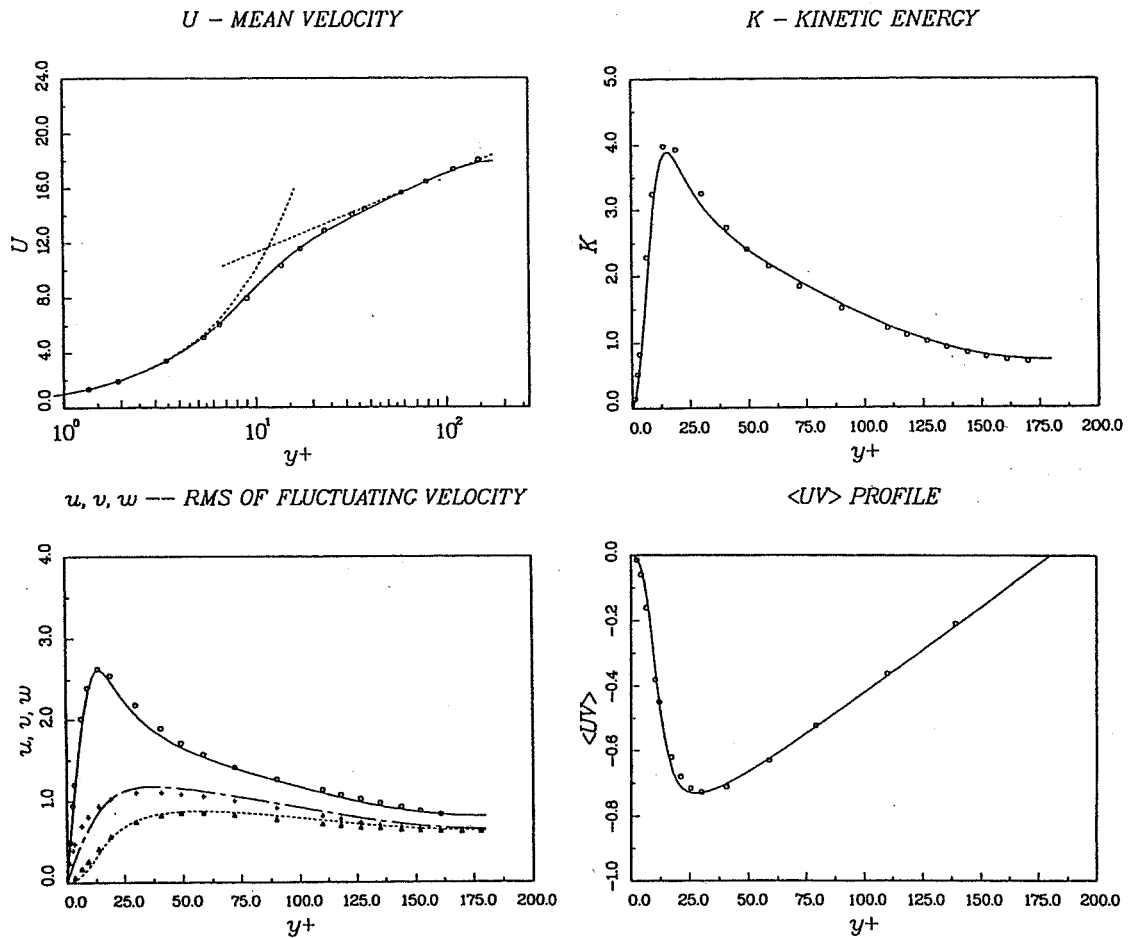
Dissipation rate equation

The modeled dissipation rate equation derived in this work is:

$$\begin{aligned}\epsilon_{,t} + U_i \epsilon_{,i} = & (\nu \epsilon_{,i} - \langle \epsilon u_i \rangle)_{,i} - \psi_0 \frac{\epsilon \tilde{\epsilon}}{\langle q^2 \rangle} \\ & - \psi_1 \frac{\tilde{\epsilon}}{\langle q^2 \rangle} \langle u_i u_j \rangle U_{i,j} - \psi_2 \frac{\nu \langle q^2 \rangle}{\epsilon} \langle u_k u_l \rangle (U_{i,jl} - U_{l,ij}) U_{i,jk}\end{aligned}$$

where

$$\begin{aligned}\psi_0 &= \frac{14}{5} + 0.98[1 - 0.33 \ln(1 - 55II)] \exp(-2.83R_t^{-1/2}) \\ \psi_1 &\approx 2.1 \\ \psi_2 &= -.15(1 - F) \\ \tilde{\epsilon} &= \epsilon - \frac{\nu \langle q^2 \rangle_{,i} \langle q^2 \rangle_{,i}}{4\langle q^2 \rangle}\end{aligned}$$

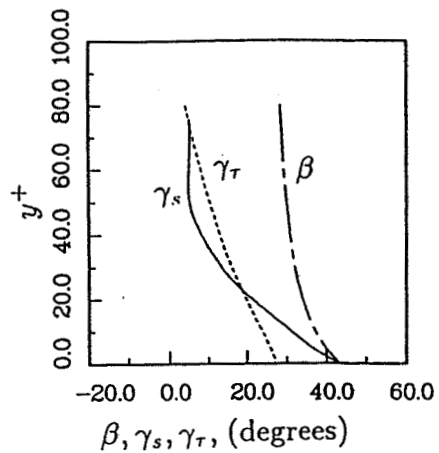
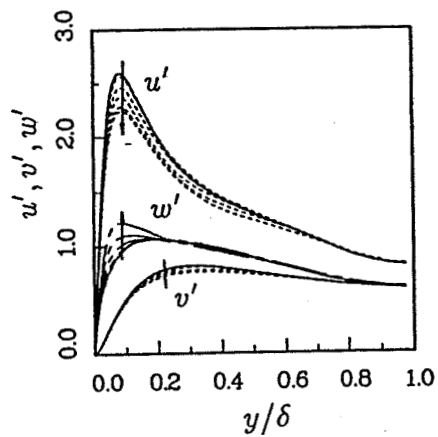
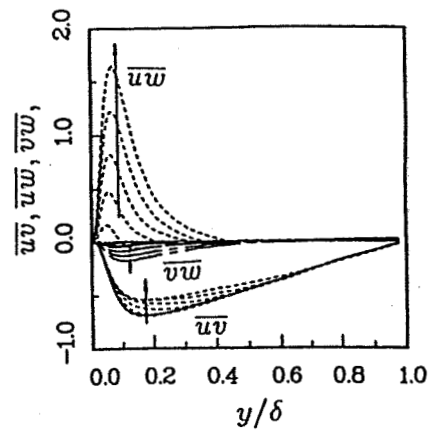
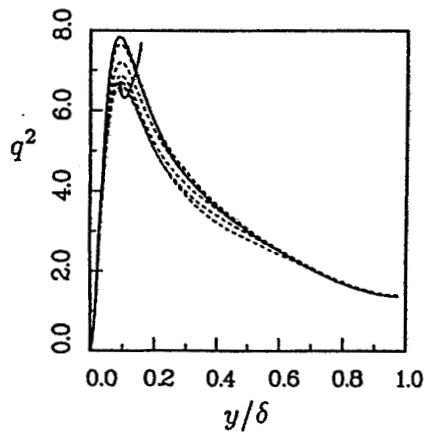
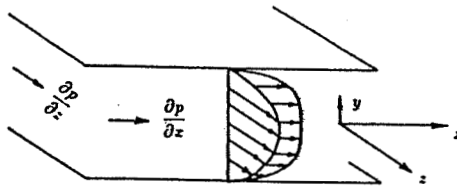
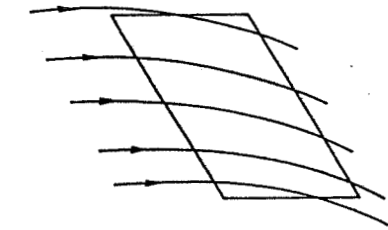


Three-dimensional boundary layer

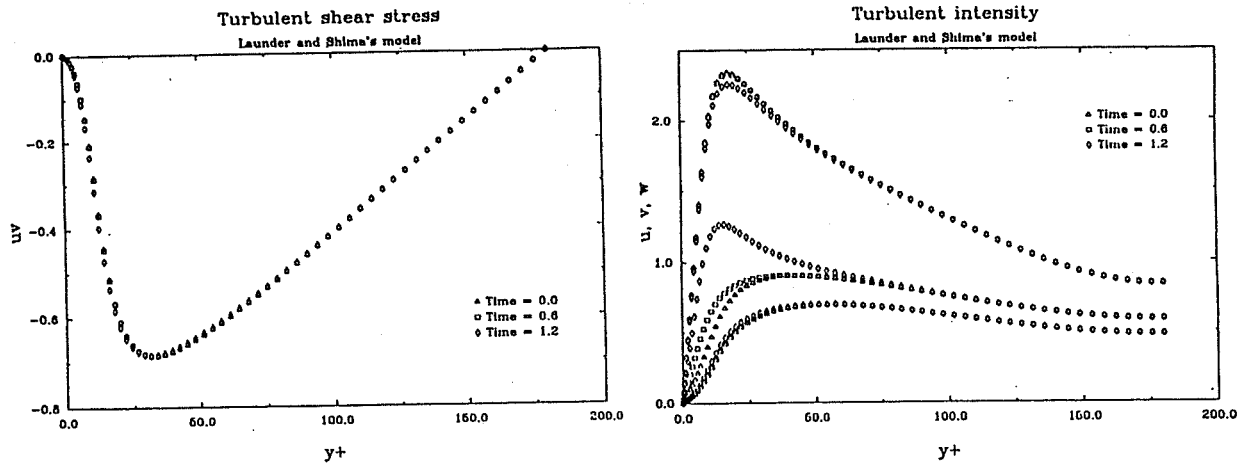
The effect of three-dimensional flow on:

- Reduction in Reynolds stress.
- Lag development between the stress and strain rate.

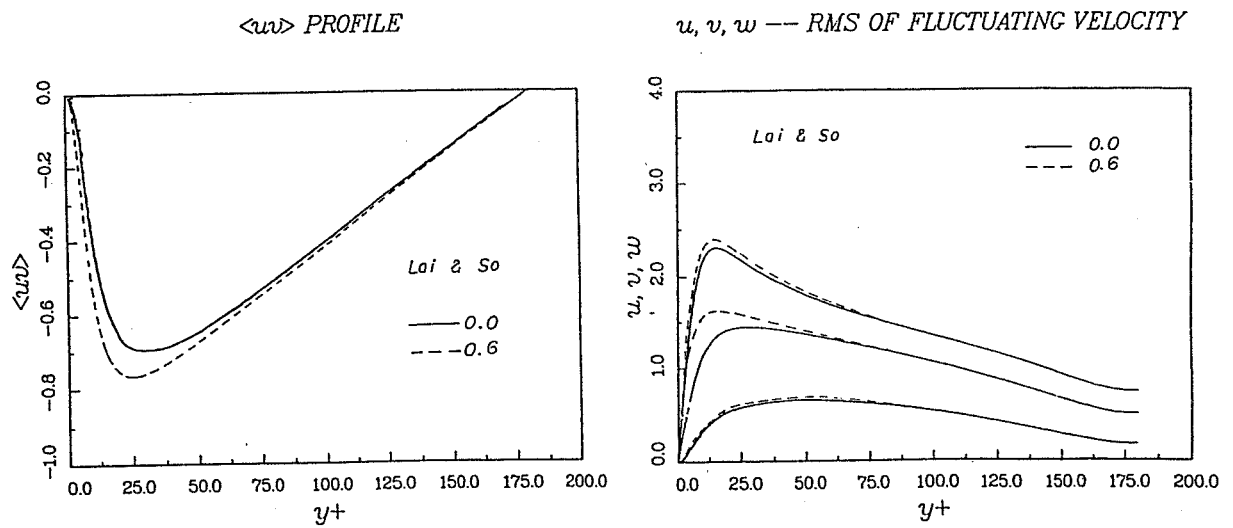
Direct numerical simulation of 3D channel flows:



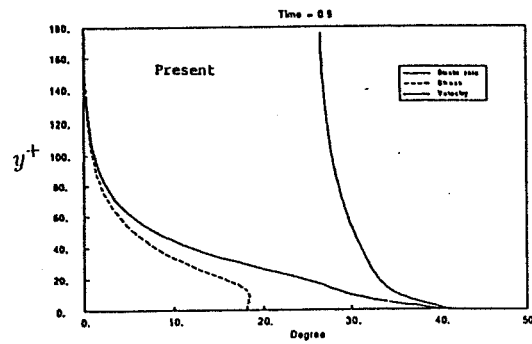
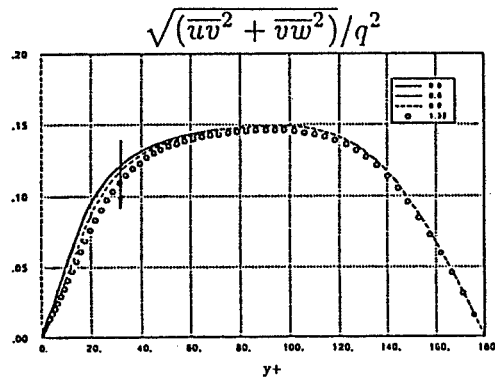
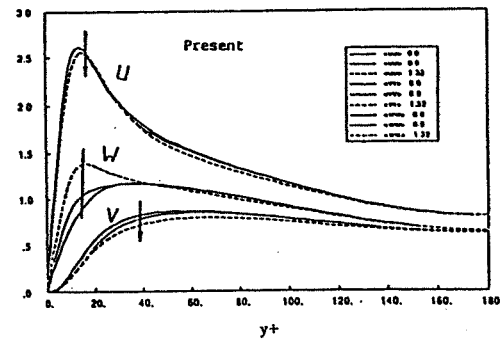
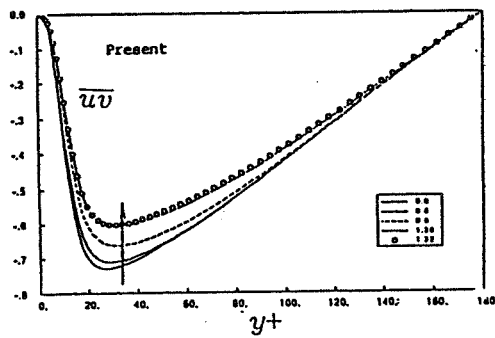
Turbulence modeling: Launder and Shima model



Turbulence modeling: Lai and So model



Turbulence modeling: Present model



Performance of existing models in modeling of 3-D flows:

- Two-equation (e.g. $k-\epsilon$) models — incapable
- Reynolds-stress models — possible

Lauder and Shima model — no

Lai and So model — no

Present model — yes

On Recontamination and Directional-Bias Problems in Monte Carlo Simulation of PDF Turbulence Models

Andrew T. Hsu
Sverdrup Technology, Inc.
Lewis Research Center Group
NASA Lewis Research Center

Turbulent combustion can not be simulated adequately by conventional moment closure turbulence models. The difficulty lies in the fact that the reaction rate is in general an exponential function of the temperature, and the higher order correlations in the conventional moment closure models of the chemical source term can not be neglected, making the applications of such models impractical. The probability density function (pdf) method offers an attractive alternative: in a pdf model, the chemical source terms are closed and do not require additional models.

The partial differential equation for the probability density function, \bar{P} , can be written as

$$\begin{aligned} \bar{\rho} \partial_t \bar{P} + \bar{\rho} \bar{v}_\alpha \partial_\alpha \bar{P} + \bar{\rho} \sum_{i=1}^N \partial_{\psi_i} \{ w_i(\psi_1, \dots, \psi_N) \bar{P} \} \\ = -\partial_\alpha (\bar{\rho} < v_\alpha'' | \psi_i > \bar{P}) - \bar{\rho} \sum_{i=1}^N \sum_{j=1}^N \partial_{\psi_i}^2 \{ \epsilon_{ij} | \psi_k > \bar{P} \} \end{aligned}$$

where the terms represent the time derivative, mean convection, chemical reaction, turbulent convection, and molecular mixing, respectively. The fact that the pdf equation has a very large dimensionality renders finite difference schemes extremely demanding on computer memory and CPU time and thus impractical, if not entirely impossible. A logical alternative is the Monte Carlo scheme, wherein the number of computer operations increases only linearly with the increase of number of independent variables, as compared to the exponential increase in a conventional finite difference scheme.

A grid dependent Monte Carlo scheme following that of J.Y. Chen and W. Kollmann has been studied in the present work. In dealing with the convection and diffusion of the pdf, the pdf equation is discretized on a given grid, e.g.,

$$\bar{P}_{x+dx,j} = \alpha_j \bar{P}_{x,j+1} + \beta_j \bar{P}_{x,j} + \gamma_j \bar{P}_{x,j-1}$$

where

$$\alpha_j + \beta_j + \gamma_j = 1$$

However, if this is the only restriction satisfied by the numerical algorithm, the mass fractions may not be conserved due to re-contamination, and directional-bias also appears. These phenomena are illustrated in Figure 1: Consider a mixing layer; use white balls to represent contaminants in the upper stream and black balls to represent contaminants in the lower stream. As the two streams move toward right, the location of the white balls and black balls are interchanged randomly to simulate convection and diffusion. From Figure 1, it is clear that directional-bias caused by recontamination caused the center of the mixing layer to drift downward. (Directional-bias can be partially corrected by changing sweeping directions.) One also notices that after the first marching step, the conservation law is violated, reflected in the Figure as missing white or black balls.

It is found that in order to conserve the mass fractions absolutely, one needs to add further restriction to the scheme, namely

$$\alpha_j + \gamma_j = \alpha_{j-1} + \gamma_{j+1}$$

A new algorithm was devised that satisfies this restriction in the case of pure diffusion or uniform flow problems. Using the same example, it is shown that absolute conservation can be achieved. This result is shown in Figure 2. One can see that the diffusion process is symmetric, and the problem of directional-bias is eliminated.

Although for non-uniform flows absolute conservation seems impossible, the present scheme has reduced the error considerably.

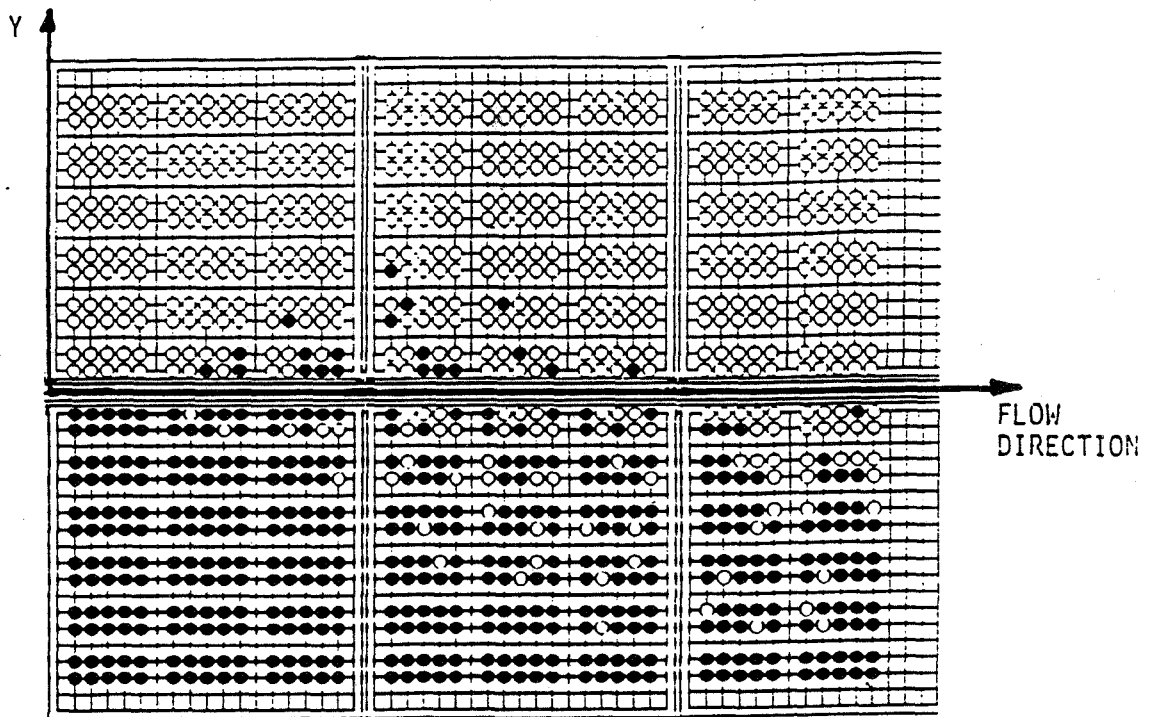


Figure 1. Illustration of directional-bias and recontamination problems in Monte Carlo simulation.

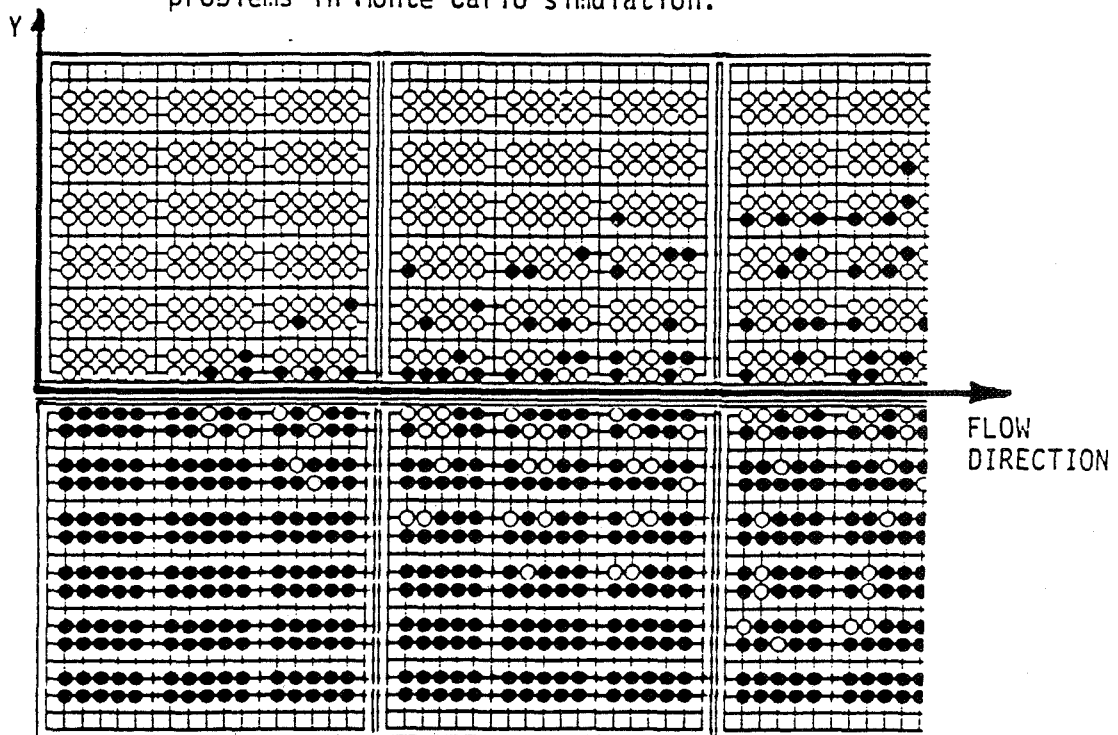


Figure 2. Solution obtained by using additional constraint in the solution procedure.

CLOSURE PROBLEM:

$\overline{u_i'' u_j''}$, - Turbulence Modeling

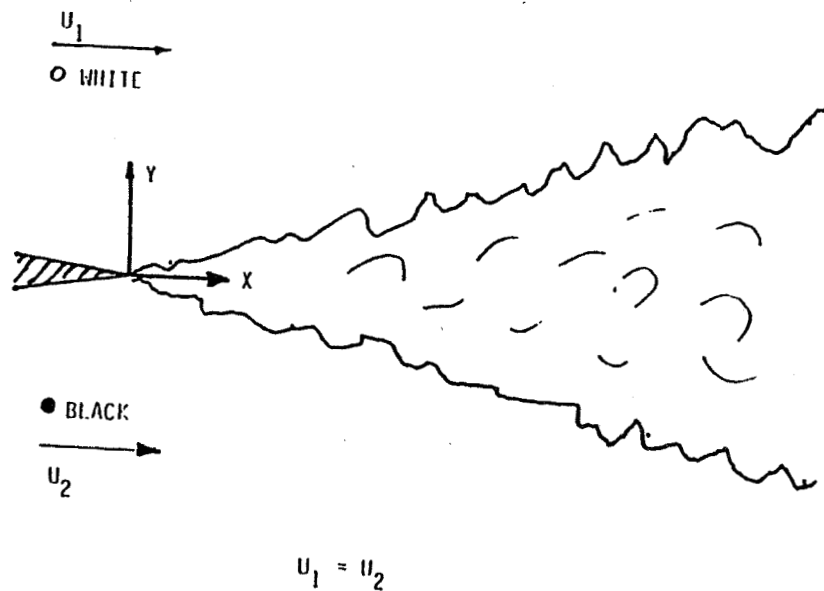
$\overline{Y_i'' u_j''}$, --- Analogy of shear stress: diffusion model.

$\rho \bar{w}_i$ --- ???

$$\rho w_i = \rho w(Y_1, \dots, Y_n, T, \rho)$$

But in general:

$$\rho \bar{w}_i \neq \rho w(\bar{Y}_1, \dots, \bar{Y}_n, \bar{T}, \bar{\rho})$$



The Simulation of Convection and Diffusion

$$\bar{\rho} \tilde{v}_a \partial_a \tilde{P} = -\partial_a (\bar{\rho} \langle v_a'' | \psi_i \rangle \tilde{P})$$

Discretize:

$$\tilde{P}^+_{x+dx,j} = \alpha_j \tilde{P}^+_{x,j+1} + \beta_j \tilde{P}^+_{x,j} + \gamma_j \tilde{P}^*_{x,j-1}$$

Where the coefficients must satisfy:

$$\alpha_j + \beta_j + \gamma_j = 1$$

MONTE CARLO METHOD

- (1) Use an ensemble of stochastic particles to represent the PDF
- (2) The particles move in such a way that the PDF evolution equation is satisfied.
- (3) Three steps: convection/diffusion; molecular mixing; chemical reaction.

PDF EVOLUTION EQUATION

$$\begin{aligned} & \bar{\rho} \partial_t \tilde{P} + \bar{\rho} \tilde{v}_\alpha \partial_\alpha \tilde{P} + \bar{\rho} \sum_{i=1}^N \partial_{\psi_i} \{ w_i(\psi_1, \dots, \psi_N) \tilde{P} \} \\ & = -\partial_\alpha (\bar{\rho} \langle v''_\alpha | \psi_i \rangle \tilde{P}) - \bar{\rho} \sum_{i=1}^N \sum_{j=1}^N \partial_{\psi_i \psi_j}^2 (\langle \epsilon_{ij} | \psi_k \rangle \tilde{P}) \end{aligned}$$

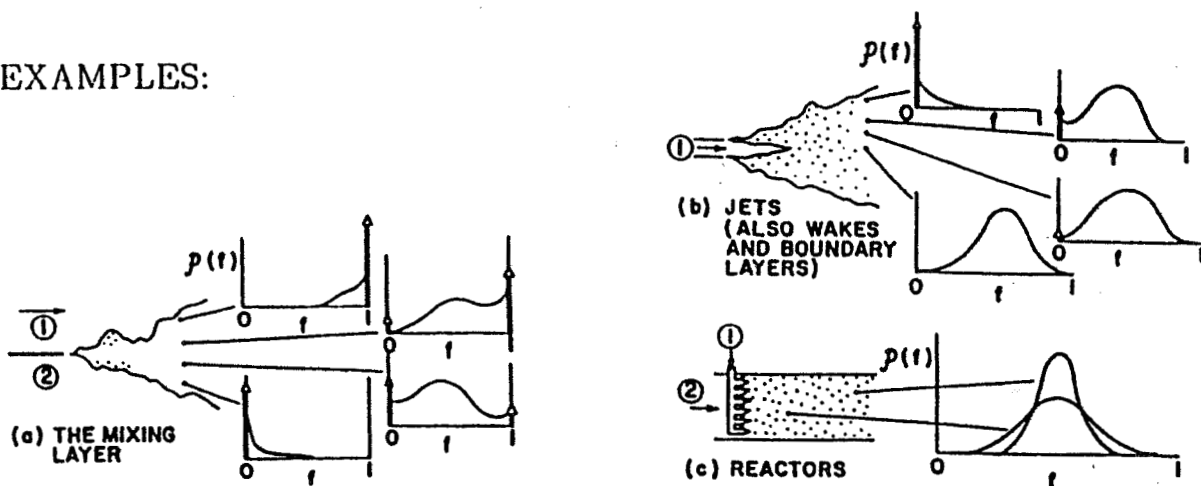
- (1) Mean Convection
- (2) Chemical Reaction
- (3) Turbulent Convection
- (4) Molecular Mixing

Dimension: $N+5$ finite difference not feasible.

PROBABILITY DENSITY FUNCTION MODEL

$$\rho \bar{w}_i = \int \dots \int \rho w_i(Y_1, \dots, Y_n, T, \rho) P(Y_1, \dots, Y_n, T, \rho) dY_1 \dots dY_n dT d\rho$$

EXAMPLES:



ASSUMED PDF

MOMENT CLOSURE MODEL

Example:

$$w_F = -BT^a \exp\left(-\frac{T_a}{T}\right) \rho^2 Y_F Y_O$$

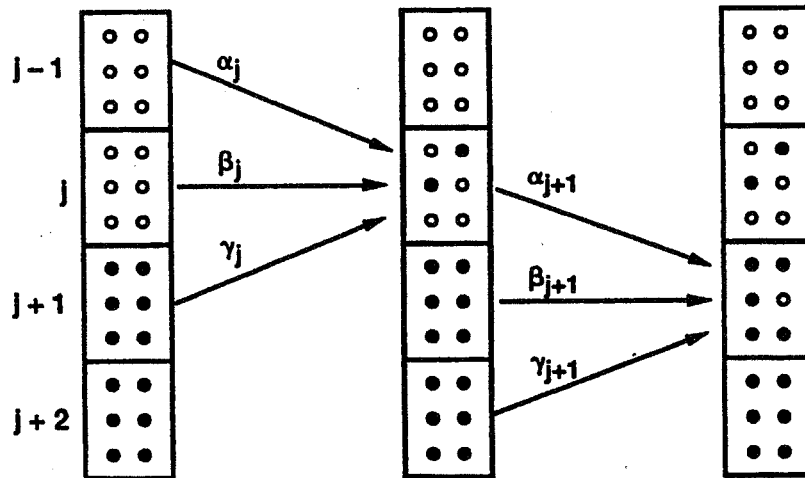
$$T = \bar{T} + T'$$

$$Y_F = \bar{Y}_F + Y'_F$$

$$Y_O = \bar{Y}_O + Y'_O$$

- w_F can be expended into series and terms such as $Y'_F \bar{Y}'_O$, $Y'_F \bar{T}'$, $Y'_O \bar{T}'$, and \bar{T}'^2 have to be modeled.

Restrictions: $T' \ll \bar{T}$, $T_a \sim \bar{T}$



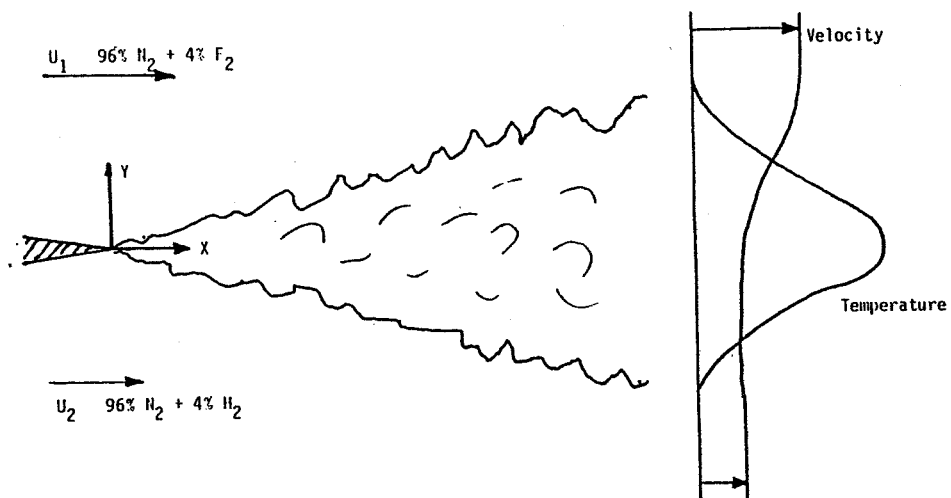
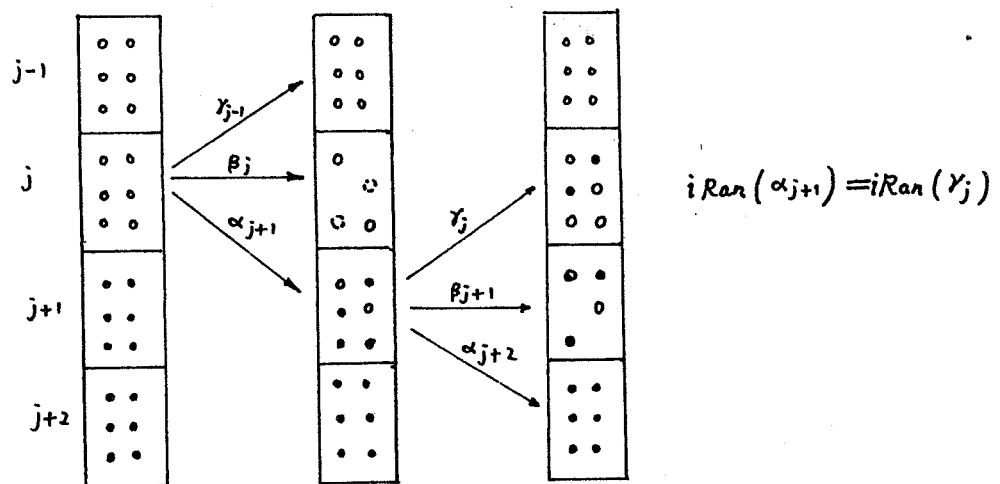
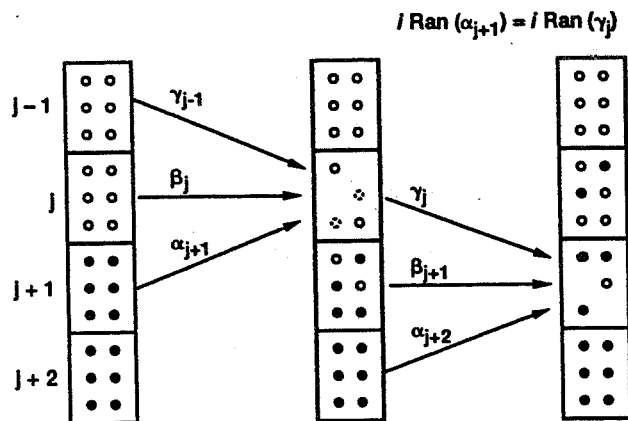


Fig.1 Velocity distribution

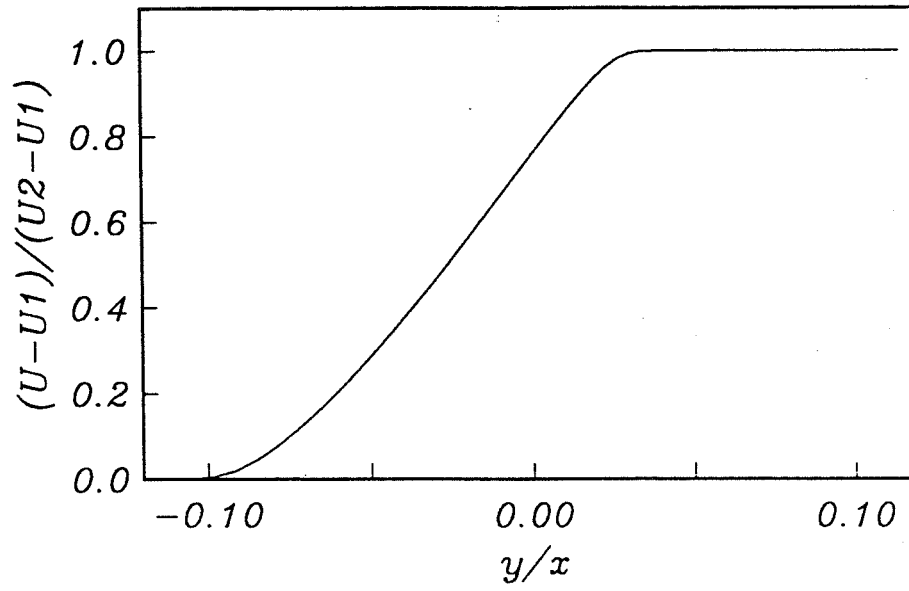


Fig.2 Turbulent kinetic energy

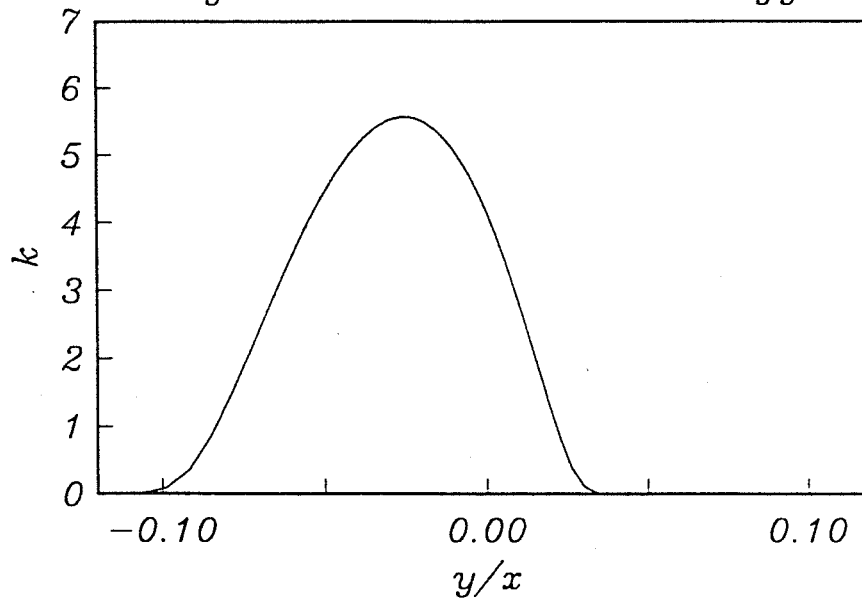


Fig.1 Temperature distribution: $H_2 + F_2 = 2HF$

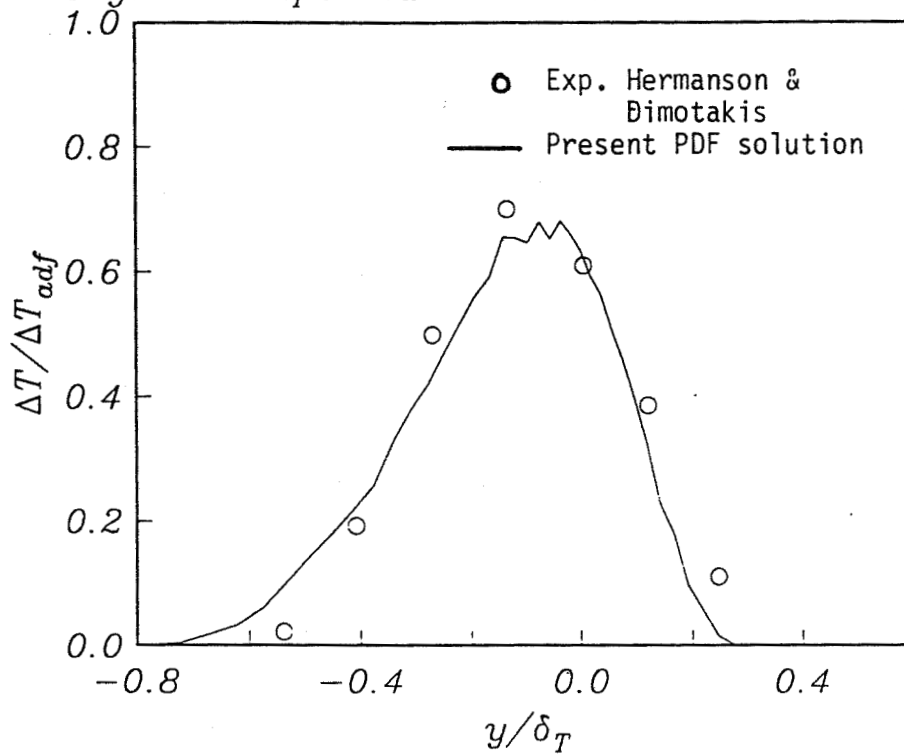
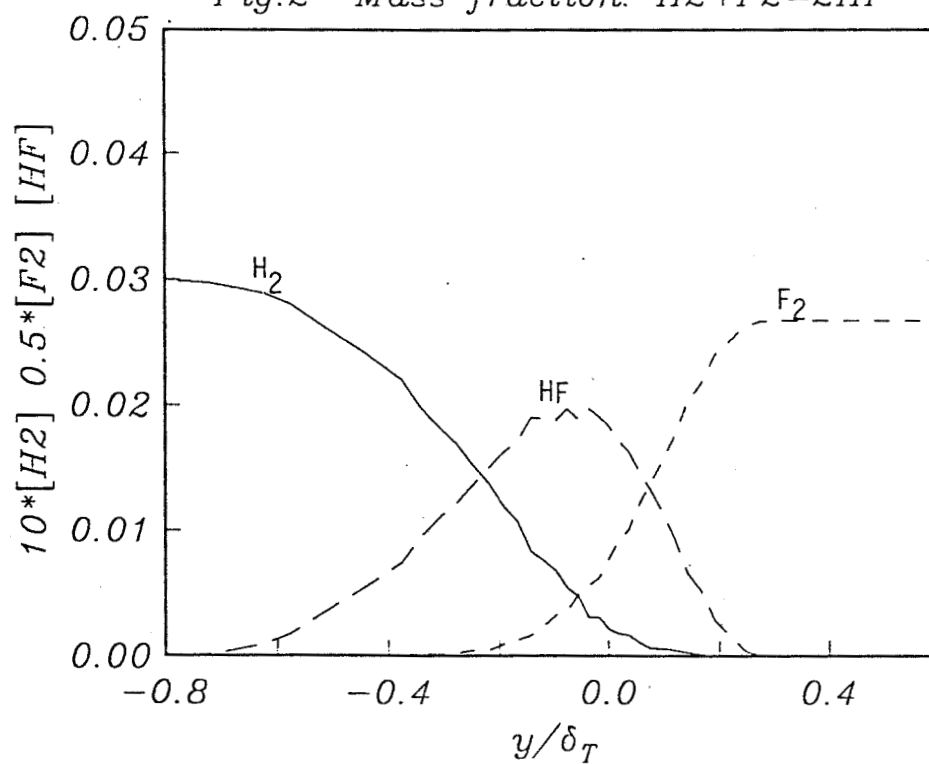


Fig.2 Mass fraction: $H_2 + F_2 = 2HF$



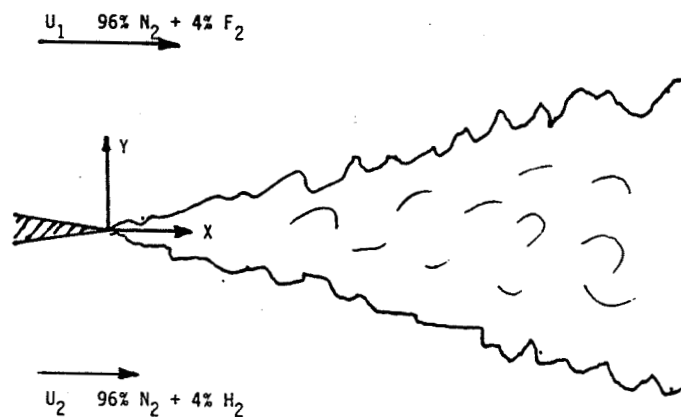
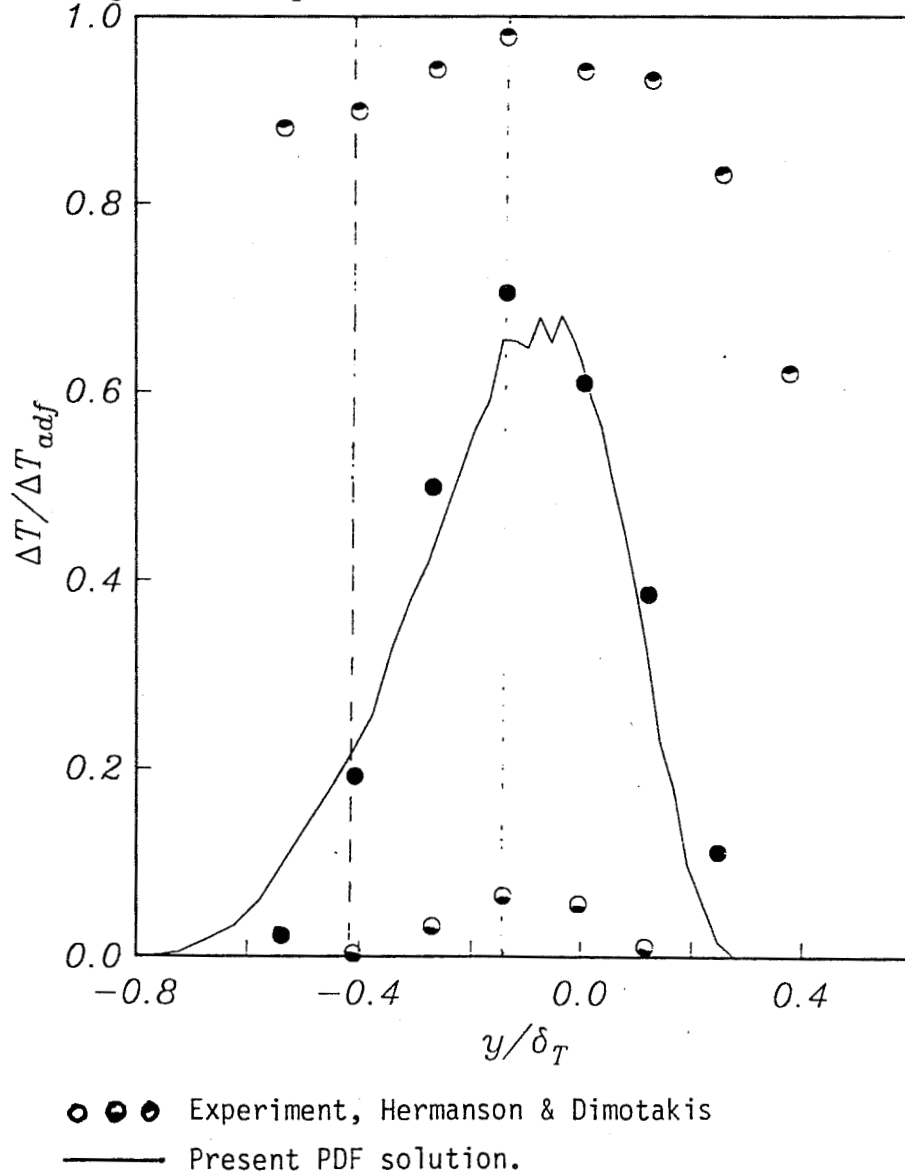


Fig.1 Temperature distribution: $H_2 + F_2 = 2HF$



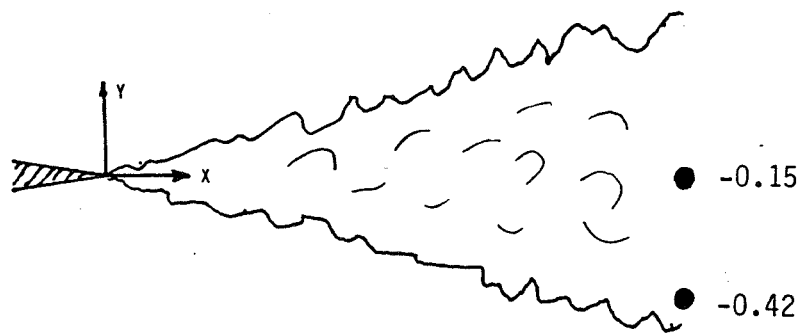
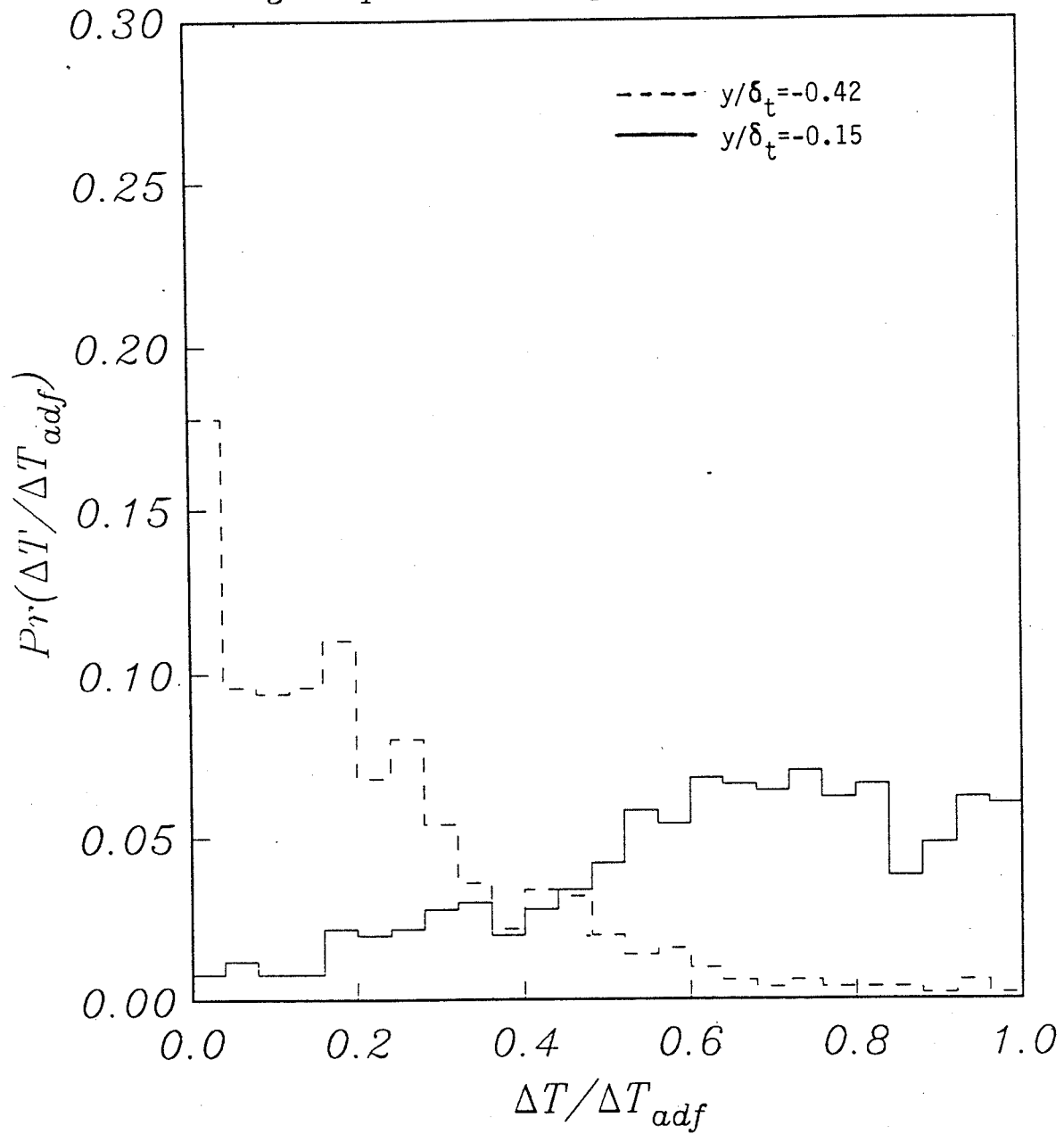


Fig.2 probability mass function.



CONCLUDING REMARKS

- (1) For absolute conservation, the number of particles received in a cell must equal to the number that are sent out from the same cell:

$$\alpha_j + \gamma_j = \alpha_{j-1} + \gamma_{j+1}$$

- (2) The particles that are replaced must be the particles that were sent out.
- (3) Store at least 3 arrays of random indices and move them successively.
- (4) Eliminated the problem of recontamination and directional-bias in the cases of uniform flow or pure turbulent diffusion. Reduced error.
- (5) In case of non-uniform flow, convection is needed, and thus (1) can not be satisfied.
- (6) The capability and accuracy of the present scheme are demonstrated through the example of a turbulent non-premixed flame.

Implementation of a $k-\epsilon$ Turbulence Model to RPLUS3D Code

Tawit Chitsomboon
Institute for Computational Mechanics in Propulsion
NASA Lewis Research Center

The RPPLUS3D code has been developed at the NASA Lewis Research Center to support the national aerospace plane project. The code has the capability to solve three dimensional flowfields with finite rate combustion of hydrogen and air. The combustion processes of the hydrogen-air system are simulated by an 18-reaction path, 8-species chemical kinetic mechanism. The code uses a Lower-Upper (LU) decomposition numerical algorithm as its basis, making it a very efficient and robust code. Except for the Jacobian matrix for the implicit chemistry source terms, there is no inversion of a matrix even though a fully implicit numerical algorithm is used.

A k -epsilon turbulence model has recently been incorporated into the RPLUS3D code. Initial validations have been conducted for a flow over a flat plate. Results of the validation studies will be shown. Some difficulties in implementing the k -epsilon equations to the RPLUS3D code will also be discussed.

MOTIVATION

- MOST APPLICATIONS ESPECIALLY FLOWS IN SCRAMJET COMBUSTORS ARE TURBULENT
- COMPLEX FLOWS IN SCRAMJET COMBUSTORS (e.g., NORMAL H_2 INJECTION INTO SUPERSONIC AIR) WILL REQUIRE AT THE MINIMUM A $k-\epsilon$ MODEL

OBJECTIVE

- IMPLEMENT A HIGH REYNOLDS NUMBER $k-\epsilon$ TURBULENCE MODEL
- A LOW REYNOLDS NUMBER MODEL IS NOT PRACTICAL FOR A 3D CHEMICALLY REACTING CFD CODE

SOME FEATURES OF RPLUS3D

- 3D CFD CODE DEVELOPED AT NASA LEWIS RESEARCH CENTER
- FULLY IMPLICIT FINITE VOLUME L-U SCHEME
- CENTRAL-DIFFERENCE SCHEME WITH JAMESON-TYPE A.V.
- FINITE-RATE HYDROGEN-AIR COMBUSTION (8-SPECIES, 18-STEP)
- EFFICIENT: NO INVERSION OF MATRICES EXCEPT FOR COMBUSTING FLOWS
- DEVELOPED ESPECIALLY FOR NASP COMBUSTORS
- COULD BE USED FOR GENERAL PROBLEMS BY SETTING VARIOUS SWITCHES

K-EPSILON TURBULENCE MODEL

- HIGH REYNOLDS NUMBER FORM
- ECONOMICAL FOR 3D COMBUSTING FLOWS
- LAW OF THE WALL
- SOLVE 2 PDE'S FOR k AND ϵ
- USE L-U ALGORITHM (THE SAME AS THE FLOW SOLVER)
- DECOUPLE FROM THE FLOW SOLVER
- MODULAR PROGRAM : ONLY ONE SUBROUTINE CALL FROM THE FLOW SOLVER

HIGH REYNOLDS NUMBER k- ϵ MODEL

- STANDARD k- ϵ MODEL

$$\frac{\partial \rho k}{\partial t} + \frac{\partial \rho k u_i}{\partial x_i} = \rho P - \rho \epsilon + \nabla \cdot \left(\mu + \frac{\mu_t}{\sigma_k} \right) \nabla k$$

$$\frac{\partial \rho \epsilon}{\partial t} + \frac{\partial \rho \epsilon u_i}{\partial x_i} = C_1 \rho P \frac{\epsilon}{k} - C_2 \rho \epsilon \frac{\epsilon}{k} + \nabla \cdot \left(\mu + \frac{\mu_t}{\sigma_\epsilon} \right) \nabla \epsilon$$

WHERE

$$P = -\overline{u_i u_j} \frac{\partial u_i}{\partial x_j}$$

AND,

$$-\overline{u_i u_j} = \nu_t (u_{i,j} + u_{j,i}) - (2/3) \delta_{ij} (k + \nu_t \nabla \cdot \vec{V})$$

BOUNDARY CONDITIONS FOR k- ϵ MODEL

- ASSUMPTIONS

- LOG LAW PROFILE

$$u^+ = (1/\kappa)\ln(Ey^+)$$

- LOCAL EQUILIBRIUM

$$\text{dissipation} = \text{production}$$

- PSEUDO CODE

- INTEGRATE FLOW EQUATIONS

- INTEGRATE k- ϵ EQUATIONS

- FIND u_τ FROM LOG-LAW PROFILE BY NEWTON'S ITERATION

- FIND k and ϵ AT THE FIRST CELL CENTER

$$k = (u_\tau)^2 / \sqrt{c_\mu} : \quad \epsilon = (u_\tau)^3 / \kappa y$$

- FIND $\nu_t = C_\mu k^2 / \epsilon$

DIFFICULTIES ENCOUNTERED

- RUN AT LOWER CFL NO. THAN LAMINAR CASES

- DUE TO EXPLICIT VISCOUS TERMS IN THE FLOW SOLVER

- IMPLICIT FORMULATION FOR BOTH CONVECTION AND DIFFUSION TERMS

- PARTIALLY IMPLICIT SOURCE TERMS

- POSITIVITY OF k- ϵ IMPOSED AUTOMATICALLY

$$\epsilon = \epsilon^n k^{n+1} / k^n$$

- REQUIRES ARTIFICIAL DAMPING

- NO DAMPING AT TWO POINTS ADJACENT TO THE WALL

LAMINAR FLOW OVER A FLAT PLATE

• MACH NO. = 0.5 ; 31 X 31 GRID

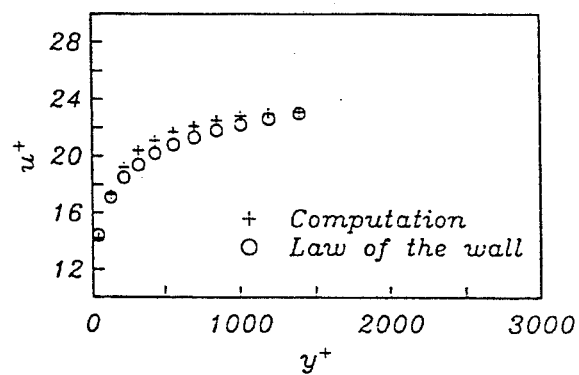
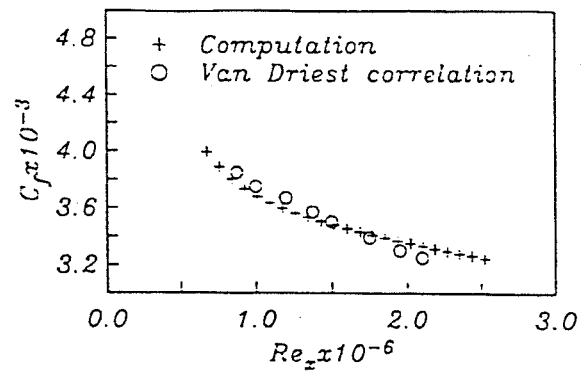
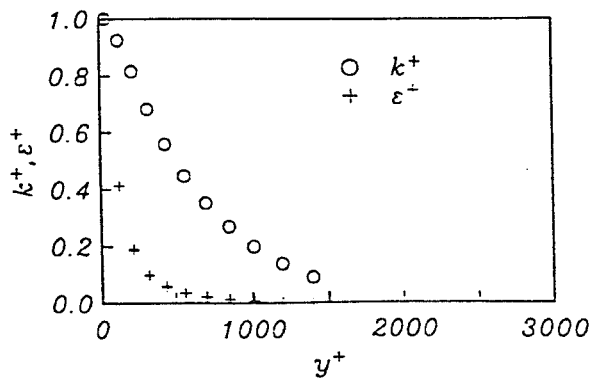
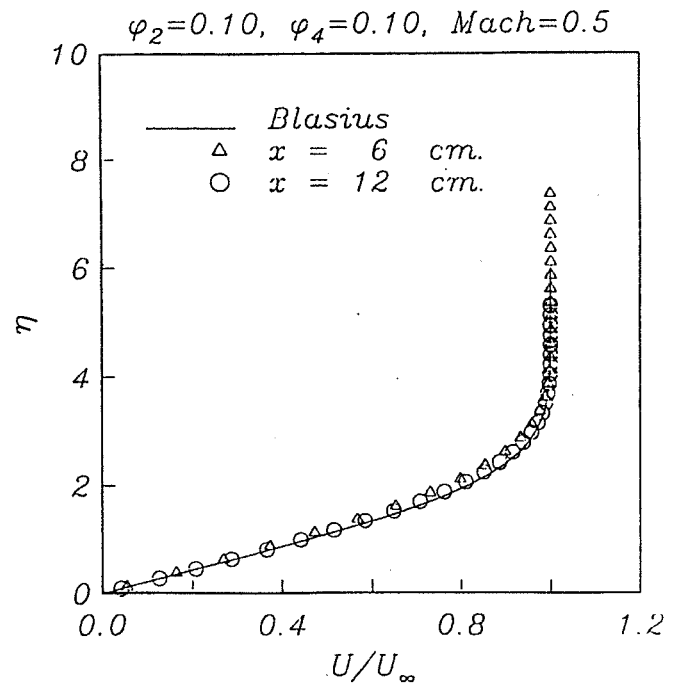


Figure . Turbulent boundary layer flow over a flat plate.

FUTURE PLANS

- MORE VALIDATIONS
 - DOUBLE NORMAL JETS IN CROSS FLOW (UVA)
 - SLANT JET IN CROSS FLOW (NASA LEWIS)
- COMPRESSIBILITY EFFECTS
- PROBABILITY DENSITY FUNCTION (PDF)

TURBULENCE AND DETERMINISTIC CHAOS

Robert G. Deissler
LeRC Research Academy
NASA Lewis Research Center

ABSTRACT

Several Turbulent and nonturbulent solutions of the Navier-Stokes equations are obtained. The unaveraged equations are used numerically in conjunction with tools and concepts from nonlinear dynamics, including time series, phase portraits, Poincaré sections, largest Liapunov exponents, power spectra, and strange attractors.

Initially neighboring solutions for a low-Reynolds-number fully developed turbulence are compared. The turbulence, which is fully resolved, is sustained by a nonrandom time-independent external force. The solutions, on the average, separate exponentially with time, having a positive Liapunov exponent. Thus, the turbulence is characterized as chaotic.

In a search for solutions which contrast with the turbulent ones, the Reynolds number (or strength of the forcing) is reduced. Several qualitatively different flows are noted. These are, respectively, fully chaotic, complex periodic (see the accompanying figure), weakly chaotic, simple periodic, and fixed-point. Of these, we classify only the fully chaotic flows as turbulent. Those flows have both a positive Liapunov exponent and Poincaré sections without pattern. By contrast, the weakly chaotic flows, although having positive Liapunov exponents, have some pattern in their Poincaré sections. The fixed-point and periodic flows are nonturbulent, since turbulence, as generally understood, is both time-dependent and aperiodic.

Besides the sustained (forced) flows, a flow which decays as it becomes turbulent is examined. As in the sustained case, the flow is extremely sensitive to small changes in initial conditions. The sensitivity increases with improved spatial resolution. For the finest grid (128^3 points) the spatial resolution appears to be quite good.

As a final note, the variation of the velocity-derivative skewness of a Navier-Stokes flow as the Reynolds number goes toward zero is calculated numerically. The value of the skewness, which has been somewhat controversial, is shown to become small at low Reynolds numbers, in agreement with intuitive arguments that nonlinear terms should be negligible.

TURBULENCE PROBABLY STILL MAIN UNSOLVED PROBLEM IN FLUID DYNAMICS.

BUT CONSIDERABLE ADVANCES BEING MADE:

- o NUMERICAL SOLUTIONS OF INSTANTANEOUS NAVIER-STOKES EQUATIONS
- o UTILIZATION OF CONCEPTS AND TOOLS FROM NONLINEAR DYNAMICS, INCLUDING DETERMINISTIC CHAOS.
"DETERMINISTIC" REFERS TO THE EVOLUTION EQUATIONS (NO RANDOM COEFFICIENTS.)

THESE TWO GO HAND IN HAND. IN FACT, USE OF NONLINEAR DYNAMICS IN STUDY OF TURBULENCE (OR IN THE STUDY OF ANYTHING) DID NOT GET FAR UNTIL ADVENT OF HIGH-SPEED COMPUTERS, BECAUSE THE NONLINEAR EVOLUTION EQUATIONS (NAVIER-STOKES EQUATIONS IN CASE OF TURBULENCE) COULD NOT PREVIOUSLY BE EFFECTIVELY USED.

HERE WE WANT TO TAKE A LOOK AT TURBULENCE BY USING CFD, TOGETHER WITH CONCEPTS AND TOOLS FROM NONLINEAR DYNAMICS.

OBTAINED SEVERAL SOLUTIONS OF INSTANTANEOUS NAVIER-STOKES EQUATIONS AT LOW REYNOLDS NUMBERS-BOTH TURBULENT AND NONTURBULENT

FLOW KEPT FROM DYING OUT BY TIME-INDEPENDENT FORCING TERM
ALTHOUGH FORCING WAS STEADY, RESULTING FLOW WAS OFTEN
HIGHLY UNSTEADY AND RANDOM IN APPEARANCE

THIS IS A REMARKABLE FEATURE OF MANY TURBULENT FLOWS - E.G., A
TURBULENT (TIME DEPENDENT) FULLY DEVELOPED PIPE FLOW MAY BE
PRODUCED BY A STEADY PRESSURE GRADIENT

SOLVED NUMERICALLY THE UNAVERAGED THREE-DIMENSIONAL
NAVIER-STOKES EQUATIONS WITH A TIME-INDEPENDENT
FORCING TERM

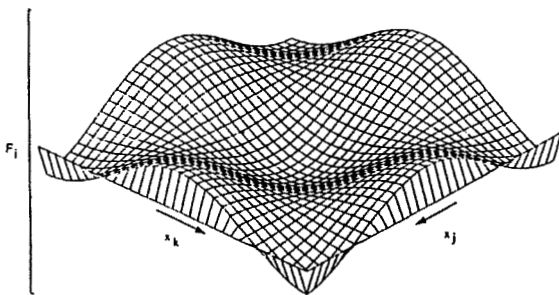
$$\frac{\partial u_i}{\partial t} = - \frac{\partial (u_i u_k)}{\partial x_k} - \frac{1}{\rho} \frac{\partial p}{\partial x_i} + \nu \frac{\partial^2 u_i}{\partial x_k \partial x_k} + F_i$$

WHERE
$$F_i = \sum_{n=1}^3 n_{a_i} n_{q_i}^2 \cos n_{\vec{q}} \cdot \vec{x}$$

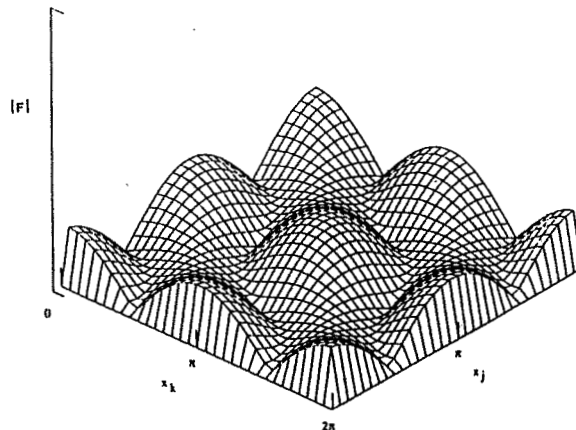
WHERE $1_{a_i} = k(2, 1, 1) \quad 2_{a_i} = k(1, 2, 1) \quad 3_{a_i} = k(1, 1, 2)$
 $1_{q_i} = (-1, 1, 1) \quad 2_{q_i} = (1, -1, 1) \quad 3_{q_i} = (1, 1, -1)$
 $Re_a = (\overline{u_1^2})^{1/2} x_0 / \nu$

AND
$$\frac{1}{\rho} \frac{\partial^2 p}{\partial x_i \partial x_i} = - \frac{\partial^2 (u_i u_k)}{\partial x_i \partial x_k}$$

PLOT OF FORCING TERM F_i ON PLANE
THROUGH GRID CENTER



MAGNITUDE OF FORCING VECTOR ON PLANE
THROUGH GRID CENTER



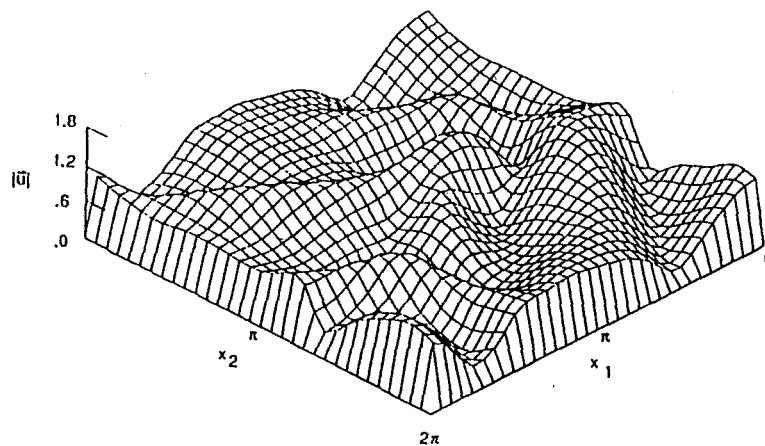
USED A CUBICAL COMPUTATIONAL GRID (0 TO 2π ON A SIDE)
WITH FOURTH-ORDER SPATIAL DIFFERENCING. 32^3 GRID POINTS

USED A PREDICTOR-CORRECTOR TIME DIFFERENCING (SECOND-ORDER LEAPFROG
PREDICTOR AND THIRD-ORDER ADAMS-MOULTON CORRECTOR)
BOUNDARY CONDITIONS: PERIODIC

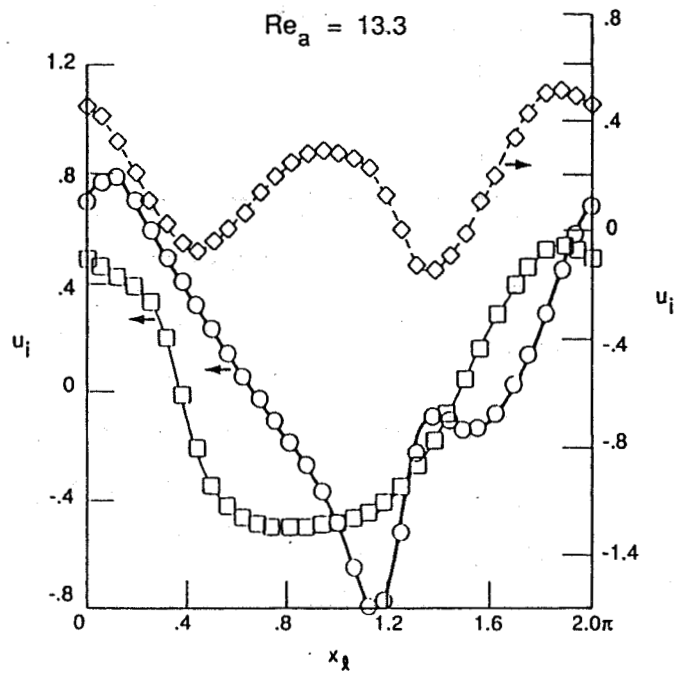
OBTAINED ASYMPTOTIC (LONG-TERM) SOLUTIONS

TO INSURE EXCITATION OF AS MANY MODES AS POSSIBLE
AT A GIVEN REYNOLDS NUMBER, INITIAL CONDITIONS ON
MOST OF FLOWS WERE SPATIALLY CHAOTIC

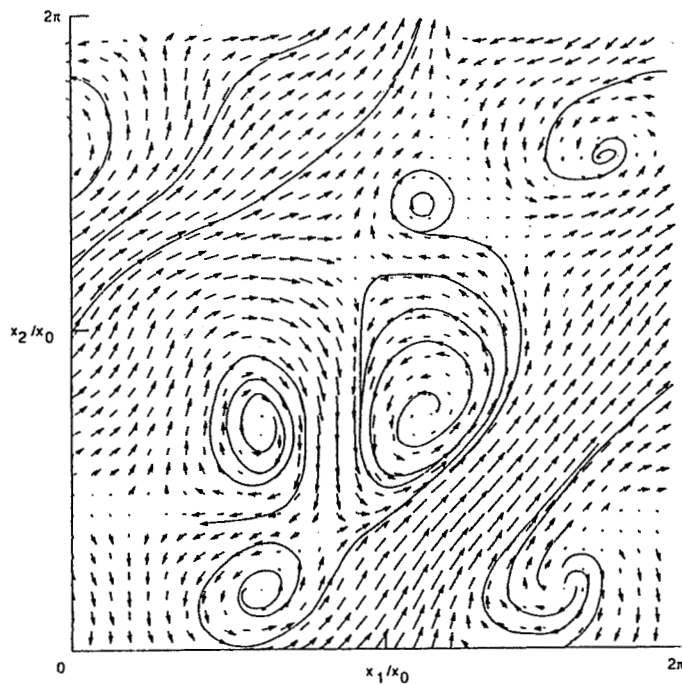
MAGNITUDE OF CHAOTIC INITIAL VELOCITY VECTOR ON PLANE THROUGH GRID CENTER



CALCULATED SPATIAL VARIATION OF VELOCITY ON PLANES THROUGH GRID CENTER



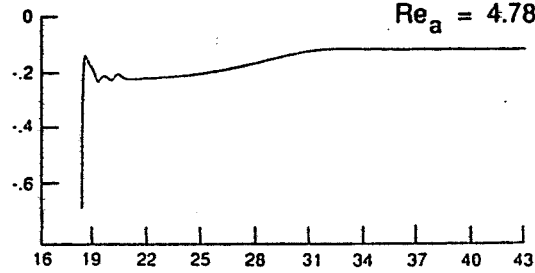
PLOT OF PROJECTION OF VELOCITY-VECTOR FIELD ON x_1 - x_2 PLANE THROUGH GRID CENTER



Time series

ASYMPTOTIC
REYNOLDS NUMBER

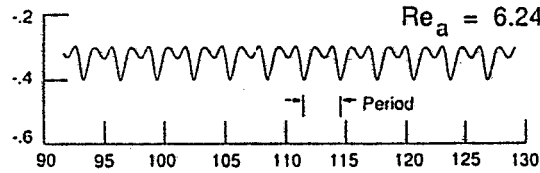
$Re_a = 4.78$



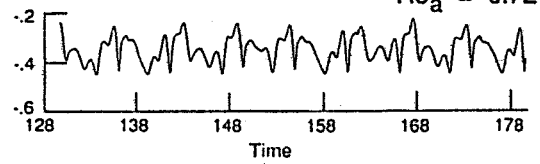
SIMPLE PERIODIC FLOW

$Re_a = 6.24$

$u_2(\pi, \pi, \pi)$



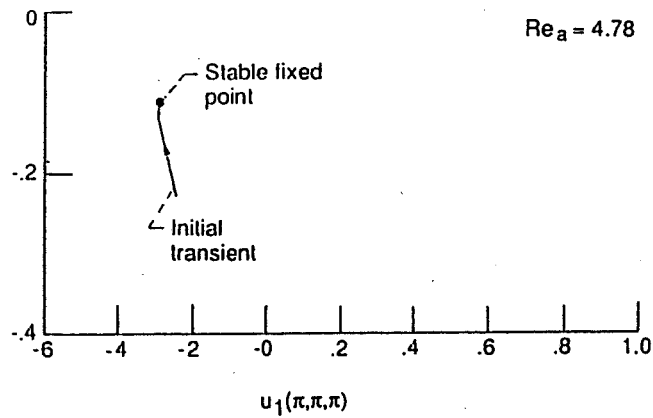
$Re_a = 6.72$

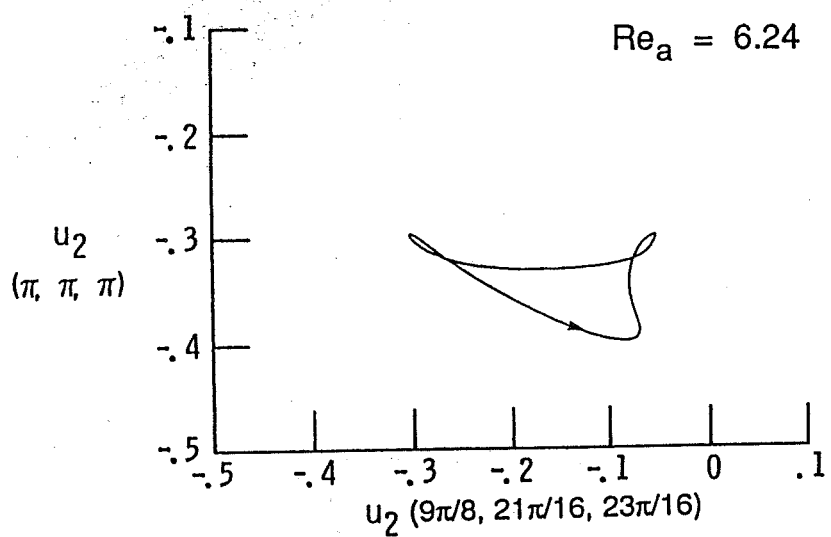
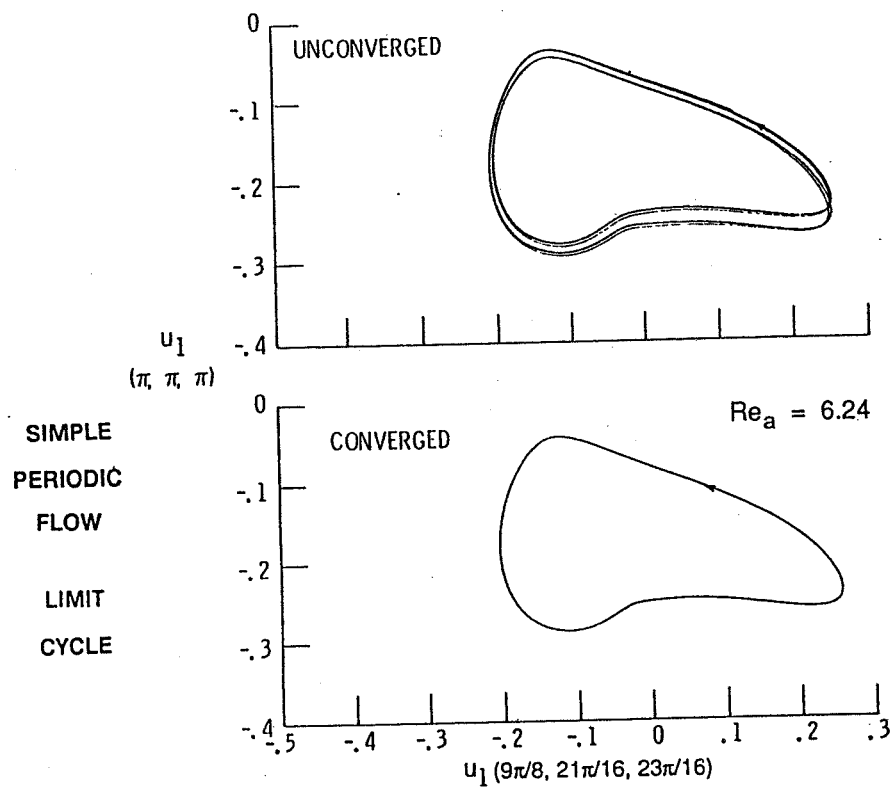


Phase portrait

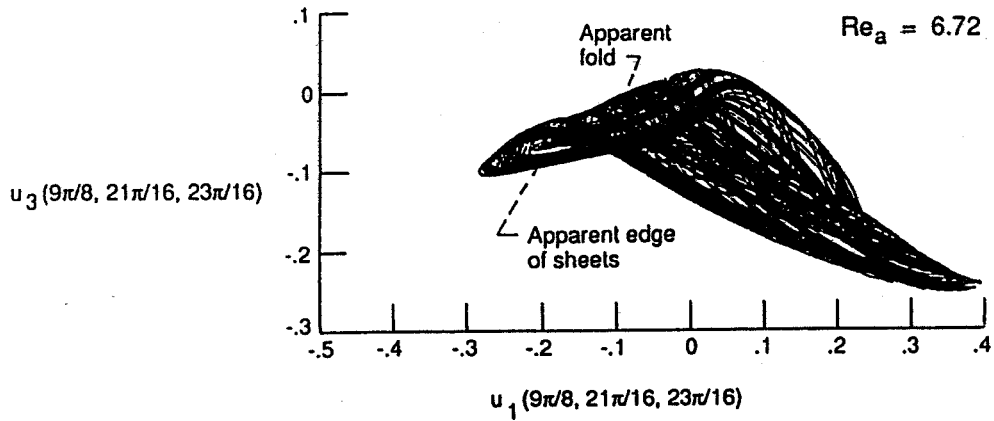
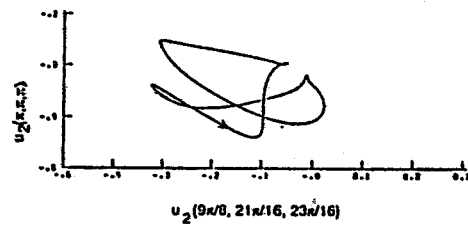
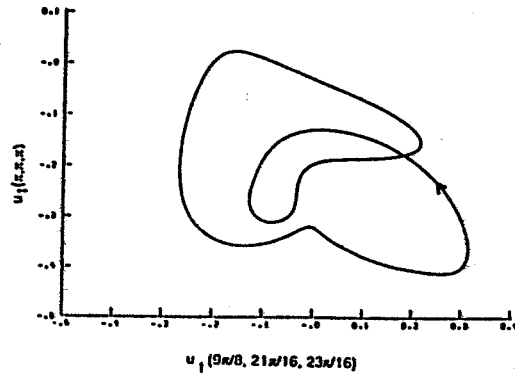
$Re_a = 4.78$

$u_2(\pi, \pi, \pi)$

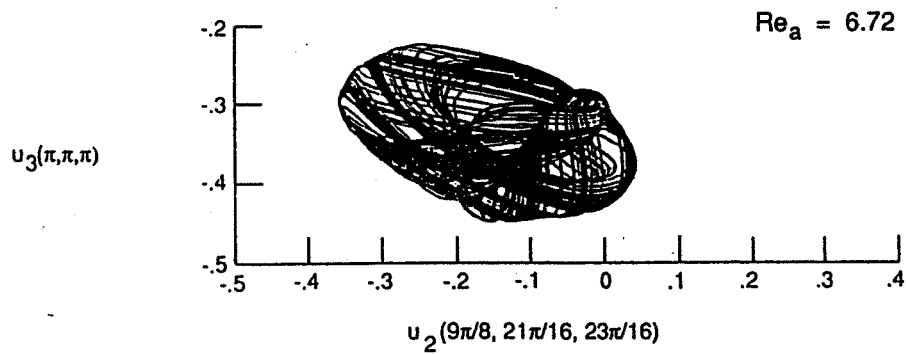


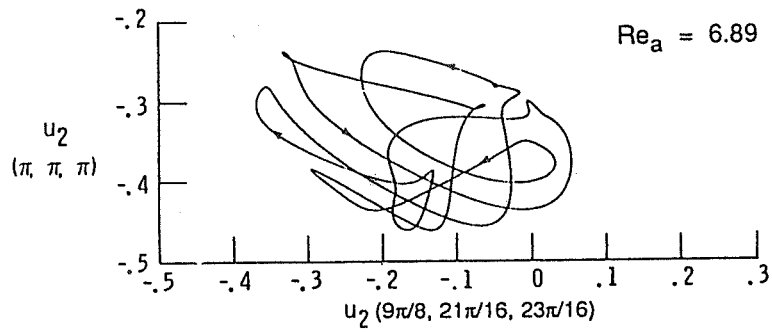


$Re_a = 6.60$

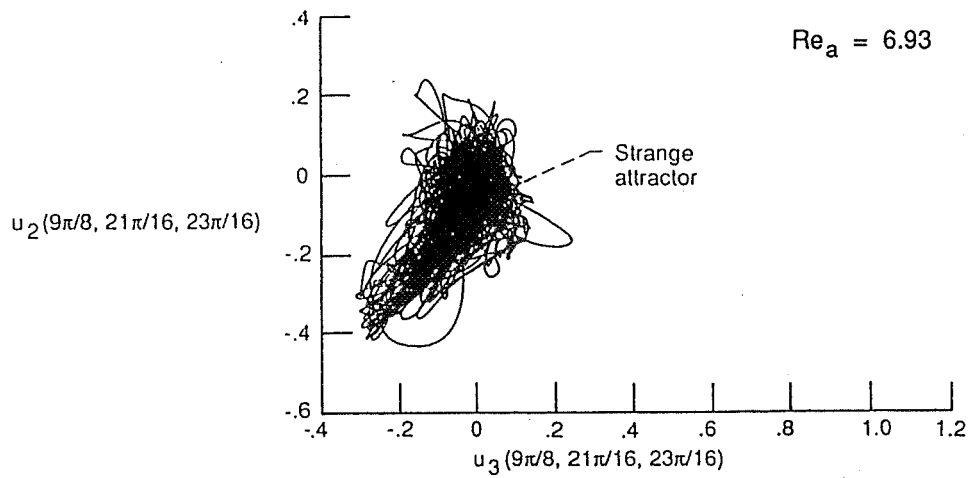


aperiodic flow

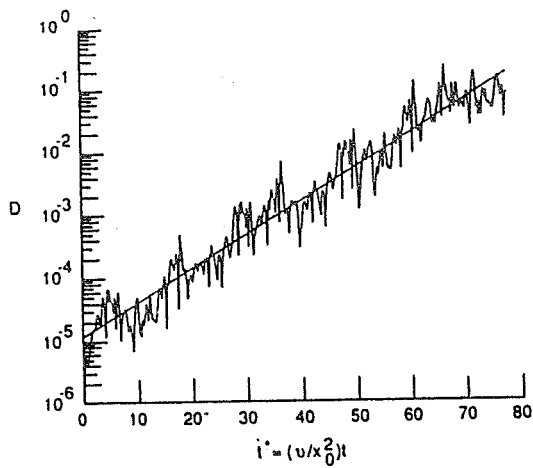




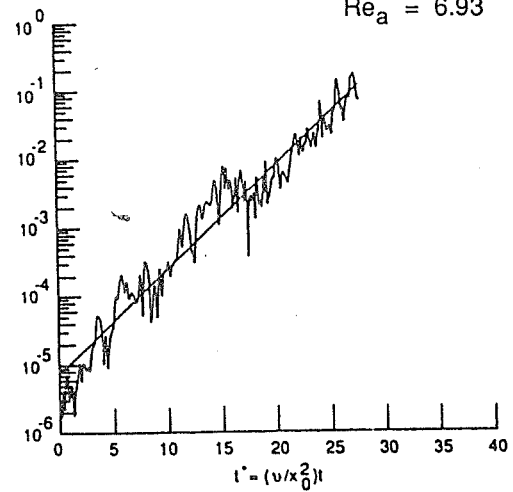
COMPLEX PERIODIC FLOW



WEAKLY CHAOTIC FLOW
 $Re_a = 6.72$



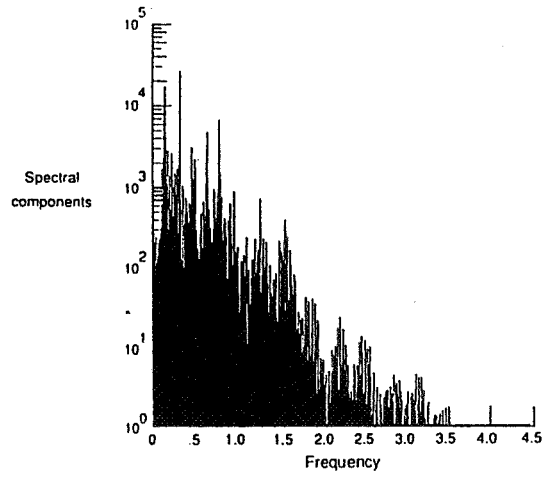
FULLY CHAOTIC FLOW
 $Re_a = 6.93$



Liapunov exponent plots
(aperiodic flows)

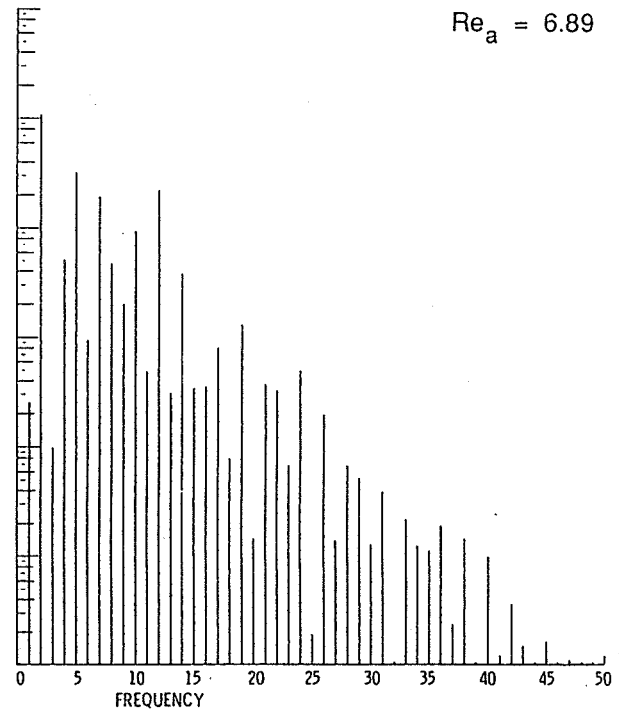
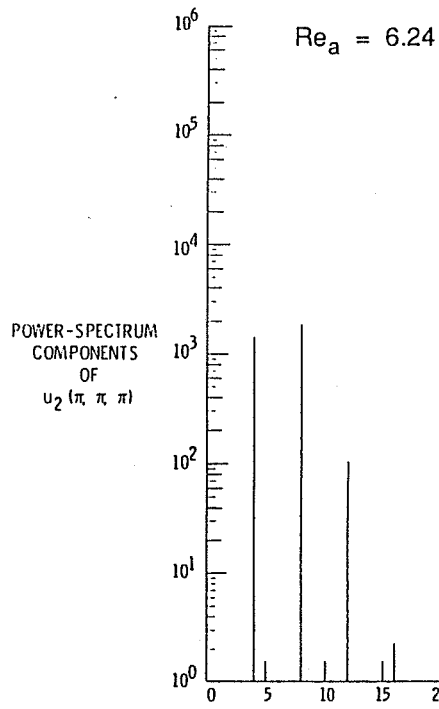
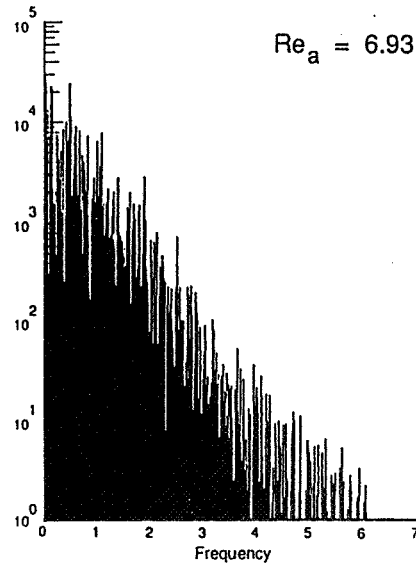
WEAKLY CHAOTIC FLOW

$$Re_a = 6.72$$



FULLY CHAOTIC FLOW

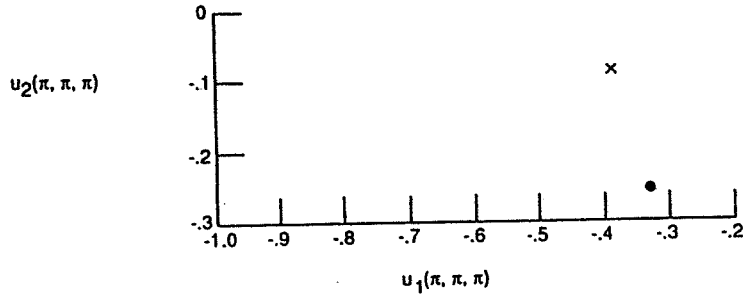
$$Re_a = 6.93$$



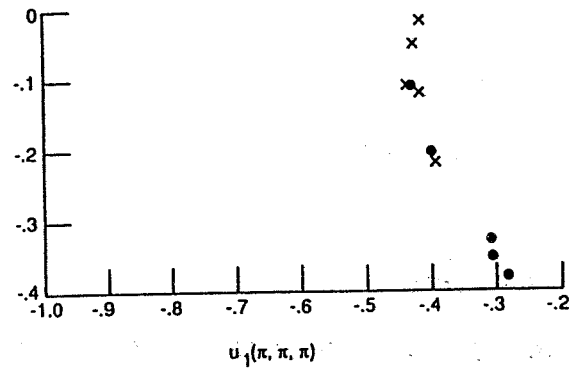
Power spectra
(Periodic flows)

Poincaré sections

$Re_a = 6.24$

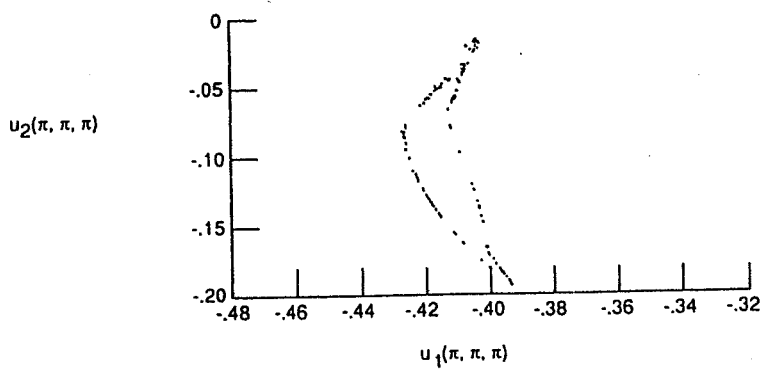


$Re_a = 6.89$

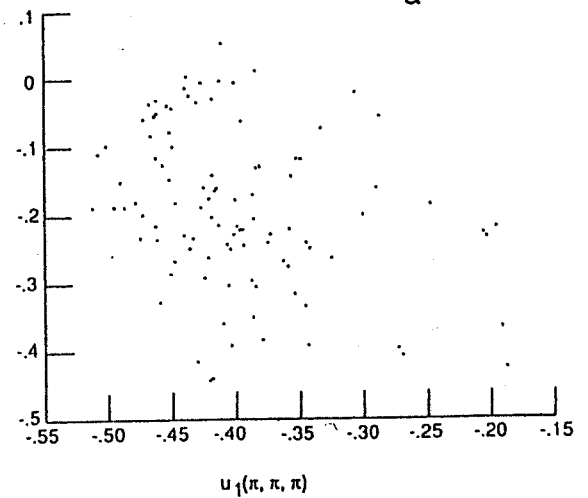


PERIODIC FLOWS

$Re_a = 6.72$



$Re_a = 6.93$



aperiodic flows

SUMMARY

FIXED POINT

LIMIT CYCLE, SIMPLE PERIODIC (PERIOD 1)

LIMIT CYCLE, COMPLEX PERIODIC (PERIOD 2)

WEAKLY CHAOTIC FLOW (LIAPUNOV EXPONENT POSITIVE, POINCARÉ SECTION HAS SOME PATTERN)

LIMIT CYCLE, COMPLEX PERIODIC (PERIOD 4) (PERIODIC WINDOW)

FULLY CHAOTIC FLOW (LIAPUNOV EXPONENT POSITIVE, POINCARÉ SECTION HAS NO APPARENT PATTERN, TURBULENT)

INCREASING
↓
REYNOLDS NUMBER

STABILITY OF COMPRESSIBLE TAYLOR-COUEFFE FLOW

by
K. Kao
Visiting Researcher
University of Colorado
and
C. Chow
University of Colorado

OBJECTIVES

- To develop both analytical and numerical tools that can be used to predict the onset of instability and subsequently to simulate the transition process by which the originally laminar flow evolves into a turbulent flow.
- To conduct the preliminary investigations with the purpose of understanding the mechanisms of the vortical structure of the compressible flow between two concentric cylinders.

INCOMPRESSIBLE TAYLOR VORTICES

- If the angular velocity of the inner cylinder is larger than a critical value, Couette flow becomes unstable so that a secondary flow starts to appear with nonvanishing radial and axial velocity components; this new flow, called Taylor-Couette flow, is in the form of opposite toroidal vortices arranged next to each other in the axial direction.

Axisymmetric laminar
Taylor vortices



COMPRESSIBLE TAYLOR VORTICES

- In the presence of heat transfer and angular motions, the influence of compressibility is nontrivially coupled and its effect on stability has yet to be determined.
- The addition of compressibility to the stability analysis greatly increases its complexity, since even in the linearized analysis there will now exist fluctuations not only in velocity and pressure but also in density, temperature, viscosity and thermal-conductivity.

GOVERNING EQUATIONS

• Mean Flow

momentum:

$$\frac{\partial P}{\partial r} = \rho \frac{w^2}{r}$$

$$\frac{\partial}{\partial r} \left[\mu \left(\frac{\partial w}{\partial r} - \frac{w}{r} \right) \right] + \frac{2\mu}{r} \left(\frac{\partial w}{\partial r} - \frac{w}{r} \right) = 0$$

energy:

$$\frac{1}{Pr} \frac{1}{r} \frac{\partial}{\partial r} \left[\kappa r \frac{\partial T}{\partial r} \right] + \mu M^2 (\gamma - 1) \left(\frac{\partial w}{\partial r} - \frac{w}{r} \right)^2 = 0 \quad M = \frac{R_1^* \Omega_1^*}{\sqrt{\gamma R^* T_1^*}}$$

• Boundary Conditions

$$\text{at } r = 1: \quad w = 1, T = 1, P = \frac{1}{\gamma M^2}, \rho = 1$$

$$\text{and at } r = \frac{R_2^*}{R_1^*}: \quad w = \frac{R_2^* \Omega_2^*}{R_1^* \Omega_1^*}, T = \frac{T_2^*}{T_1^*} = \vartheta$$

GOVERNING EQUATIONS

• Small Perturbation

$$\phi(\xi, \eta, \zeta, t) = \Phi + \bar{\phi}(\eta) \exp[i(\alpha \xi + \beta \zeta - \omega t)]$$

where

- ξ, η, ζ : computational coordinates in axial, radial and circumferential directions, respectively
- Φ : steady unperturbed flow,
 $= [U_o, V_o, W_o, P_o, T_o, \rho_o, \mu_o, \lambda_o, \kappa_o]^T$
- $\bar{\phi}$: complex disturbances
- α : disturbance wave number in ξ direction
- β : disturbance wave number in ζ direction
- ω : complex frequency

• Boundary Conditions for Disturbances

$$\begin{aligned} & \text{at } r = 1 : \quad \tilde{u}, \tilde{v}, \tilde{w}, \tilde{\tau} = 0 \\ & \text{and at } r = \frac{R_2^*}{R_1^*} : \quad \tilde{u}, \tilde{v}, \tilde{w}, \tilde{\tau} = 0 \end{aligned}$$

Note that no boundary conditions are required for pressure fluctuations when the equations are solved on a pressure staggered mesh.

METHOD OF SOLUTION

- Numerical solutions for the mean flow
- Chebyshev collocation spectral method for stability analysis

Momentum and Energy Equations:

Gauss-Lobatto points for $\tilde{u}, \tilde{v}, \tilde{w}, \tilde{\tau}$

Continuity Equation:

Gauss points for \tilde{p}

Momentum and Energy Equations:

$$\begin{aligned} & \mathbf{A}_{GL} \mathbf{L}_{GL}^2 \phi + \mathbf{B}_{GL} \mathbf{L}_{GL} (\phi + \mathbf{I}_G^{GL} \mathbf{P}) + \mathbf{C}_{GL} (\phi + \mathbf{I}_G^{GL} \mathbf{P}) \\ & = \omega \left[\mathbf{D}_{GL} \mathbf{L}_{GL} (\phi + \mathbf{I}_G^{GL} \mathbf{P}) + \mathbf{E}_{GL} (\phi + \mathbf{I}_G^{GL} \mathbf{P}) \right] \end{aligned}$$

Continuity Equation:

$$\mathbf{B}_G \mathbf{L}_G \mathbf{I}_{GL}^G \phi + \mathbf{C}_G (\mathbf{I}_{GL}^G \phi + \mathbf{P}) = \omega \mathbf{E}_G (\mathbf{I}_{GL}^G \phi + \mathbf{P})$$

where

$$\phi = \begin{pmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \\ \tilde{\tau} \end{pmatrix}, \quad \mathbf{P} = (\tilde{p})$$

and

- $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}, \mathbf{E}$: coefficient matrices
 \mathbf{L} : spectral differentiation operator
 \mathbf{I} : spectral interpolation matrix

VERIFICATION

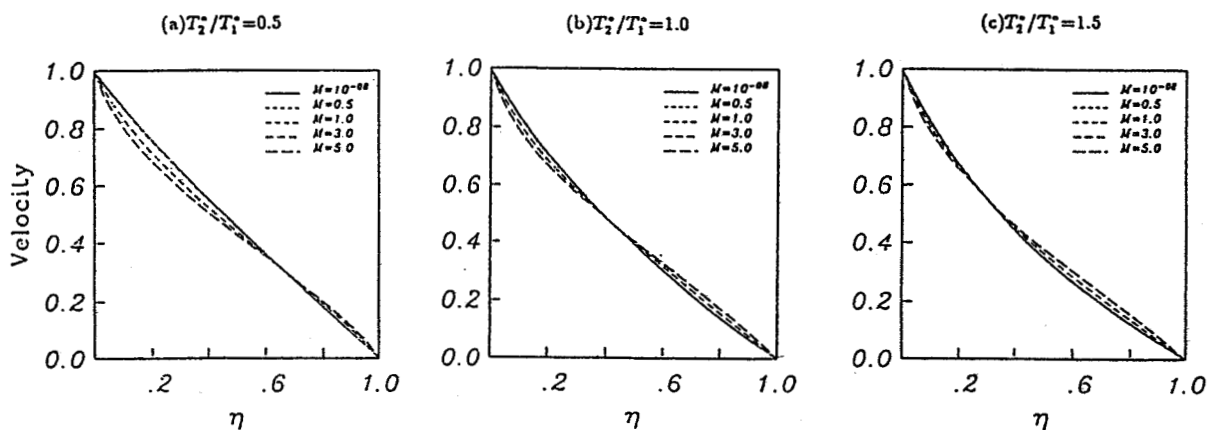
Comparison of the Critical Reynolds Number for Temporal Stability of Taylor-Couette Flow ($R_1^/R_2^*=0.5$)*

	Sparrow <i>et al.</i> ²⁰	Present
Ω_2^*/Ω_1^*	Re_c	Re_c
0.0000	68.19	68.187
0.0600	73.02	73.422
0.1250	84.30	84.346
0.1800	107.21	106.923
0.2350	221.33	220.867
-0.10	66.18	66.172
-0.20	70.03	69.895
-0.30	80.93	80.418
-0.35	88.81	89.947
-0.50	114.24	115.025

RESULT AND DISCUSSION

MEAN VELOCITY PROFILES

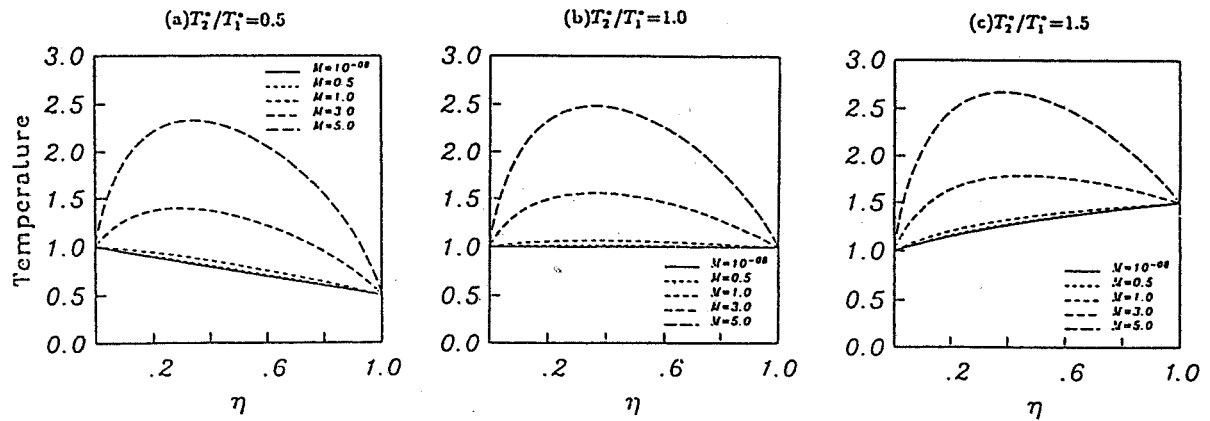
$R_1^*/R_2^*=0.5$ and $\Omega_2^*=0$



RESULT AND DISCUSSION

MEAN TEMPERATURE PROFILES

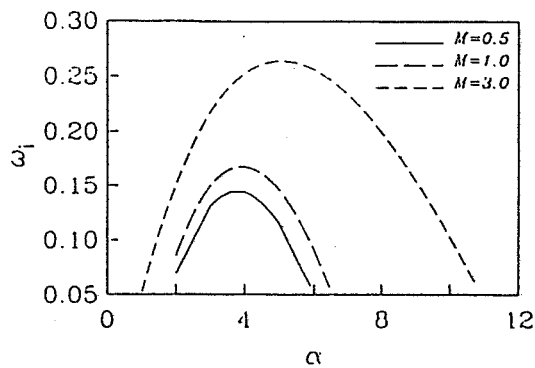
$$R_1^*/R_2^*=0.5 \text{ and } \Omega_2^*=0$$



RESULT AND DISCUSSION

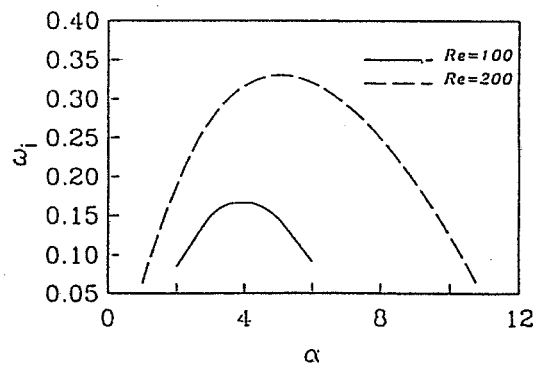
EFFECT OF MACH NUMBER

$$Re=100 \text{ and } \vartheta=1.0$$



EFFECT OF REYNOLDS NUMBER

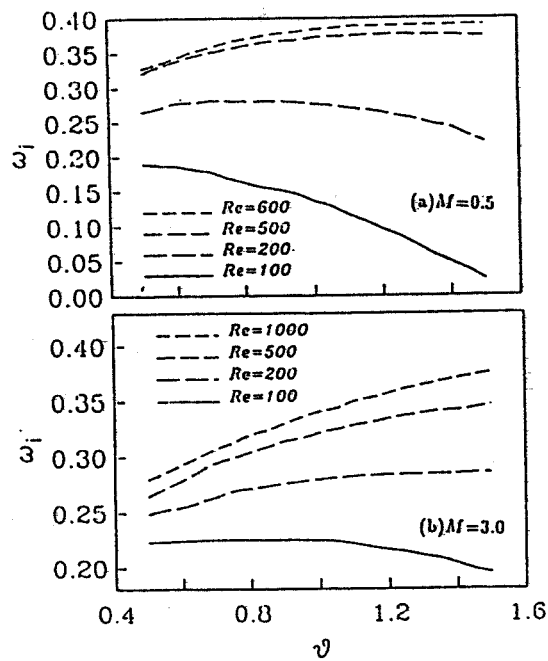
$$M=1.0 \text{ and } \vartheta=1.0$$



RESULT AND DISCUSSION

EFFECT OF TEMPERATURE RATIO

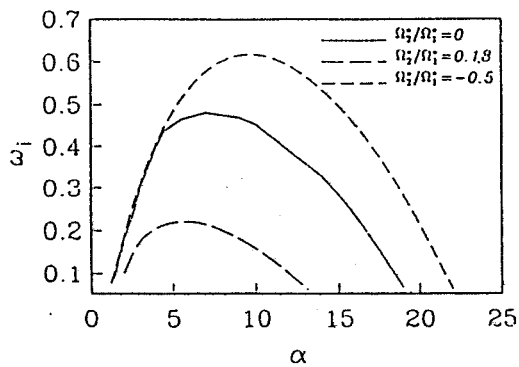
$$\alpha = 3.162, \Omega_2^* = 0$$



RESULT AND DISCUSSION

EFFECT OF SPIN MOTION

$$M=1.0 \text{ and } Re=500$$

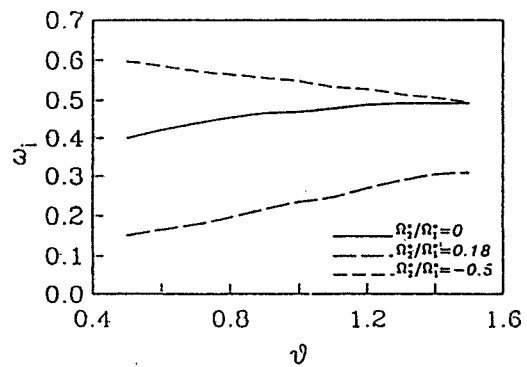


MAXIMUM GROWTH RATE

VS

TEMPERATURE RATIO

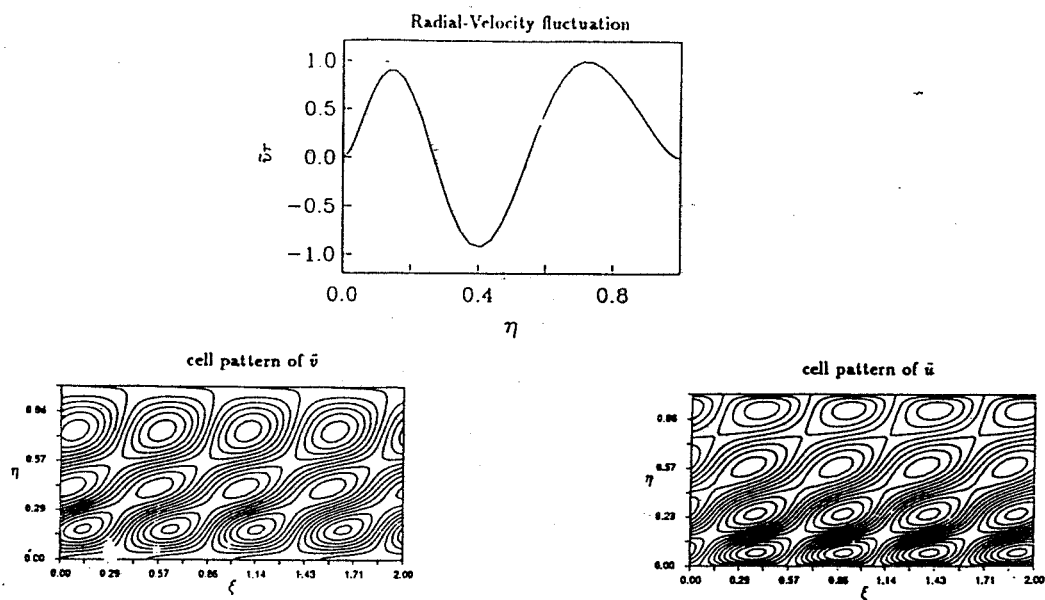
$$\alpha = 6.2, M=1.0, \text{ and } Re=500$$



RESULT AND DISCUSSION

$$M=1.0, Re=800, \Omega_2^*/\Omega_1^*=0.0$$

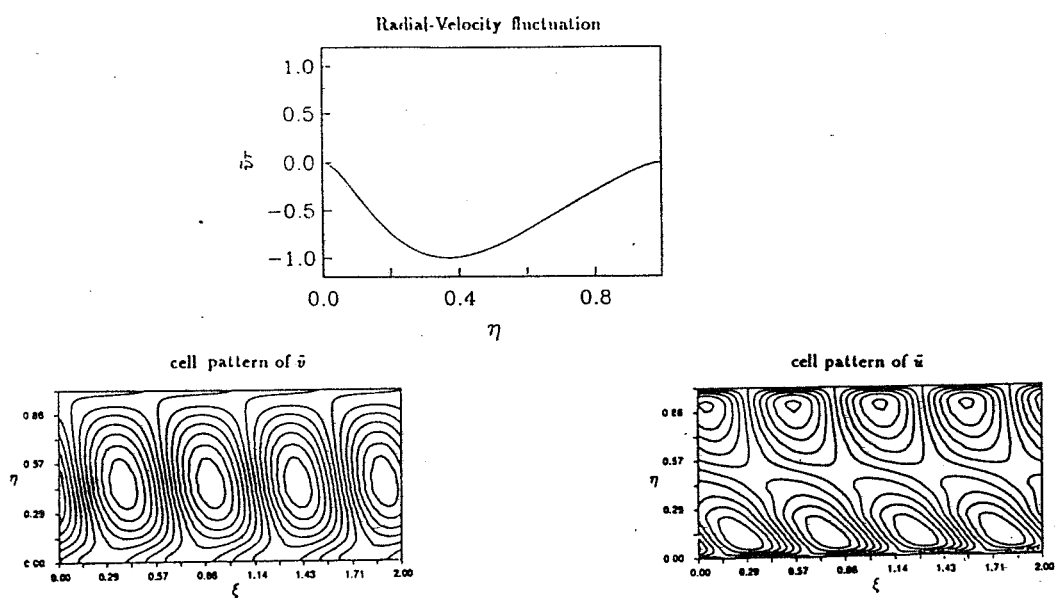
$$\alpha=6.2, \text{ and } \vartheta=1.0$$



RESULT AND DISCUSSION

$$M=1.0, Re=800, \Omega_2^*/\Omega_1^*=0.18$$

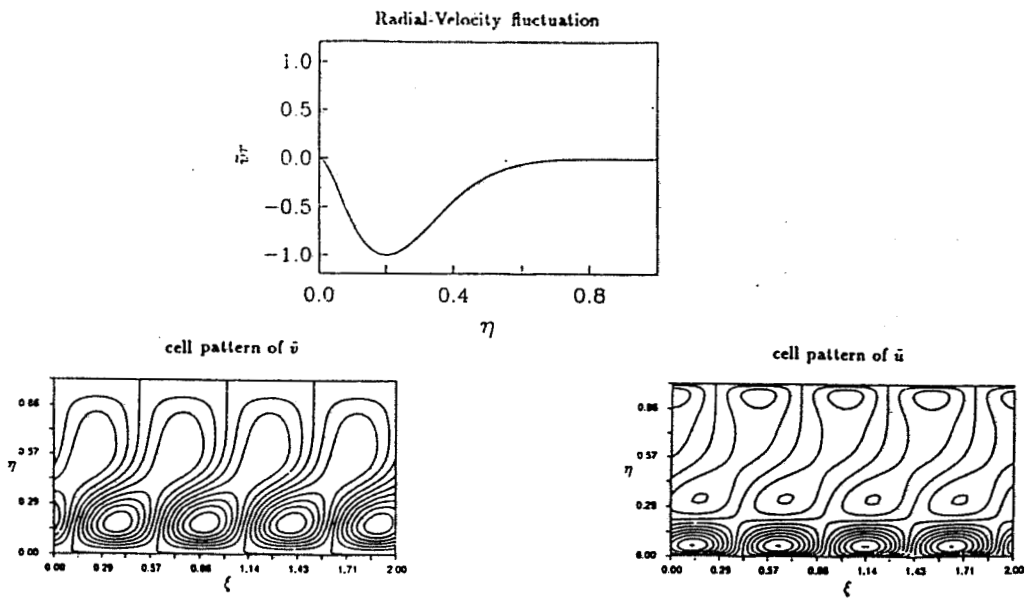
$$\alpha=6.2, \text{ and } \vartheta=1.0$$



RESULT AND DISCUSSION

$$M=1.0, Re=800, \Omega_2^*/\Omega_1^*=-0.5$$

$$\alpha=6.2, \text{ and } \vartheta=1.0$$



CONCLUDING REMARKS

- For the case of rotating inner cylinder, the higher Mach number flows are unstable to a larger range of wave numbers.
- By increasing the temperature ratio, the flow between counter rotating cylinders is stabilized with the growth rate decaying in a linear fashion. However, the effect of increasing temperature ratio is to amplify the maximum growth rate in the flow with a positive rotating speed ratio.
- Further examinations can be made on those effects of axial flow, gap width, finite axial length, and other possible influential factors.

Techniques for Animation of CFD Results

Dr. Jay Horowitz

Computer Science Division
NASA Lewis Research Center

and

Jeffery C. Hanson

Sverdrup Technology, Inc.
Lewis Research Center Group
NASA Lewis Research Center

Intro

Video animation is becoming increasingly vital to the CFD researcher, not just for the presentation results, but for recording and comparing dynamic visualizations that are beyond the current capabilities of even the most powerful graphic workstations. To meet these needs Lewis Research Center has recently established a facility to provide users with easy access to advanced video animation capabilities. However, producing animation that is both visually effective and scientifically accurate involves various technological and aesthetic considerations that must be understood both by the researcher and by those supporting the visualization process. Many of these considerations will be addressed in the presentation.

Scan Conversion

Conversion of high-resolution workstation images to low-resolution television images is performed by hardware that either converts only a portion of the screen, or invokes various pixel-averaging techniques. The former can result in excessive aliasing, whereas the latter can result in smearing and disappearance of points and lines. Both can have trouble displaying semi-transparent objects from workstation using bit-mask transparency algorithms. Scan conversion is a major problem in accurately portraying Computational Fluid Dynamics grid geometry.

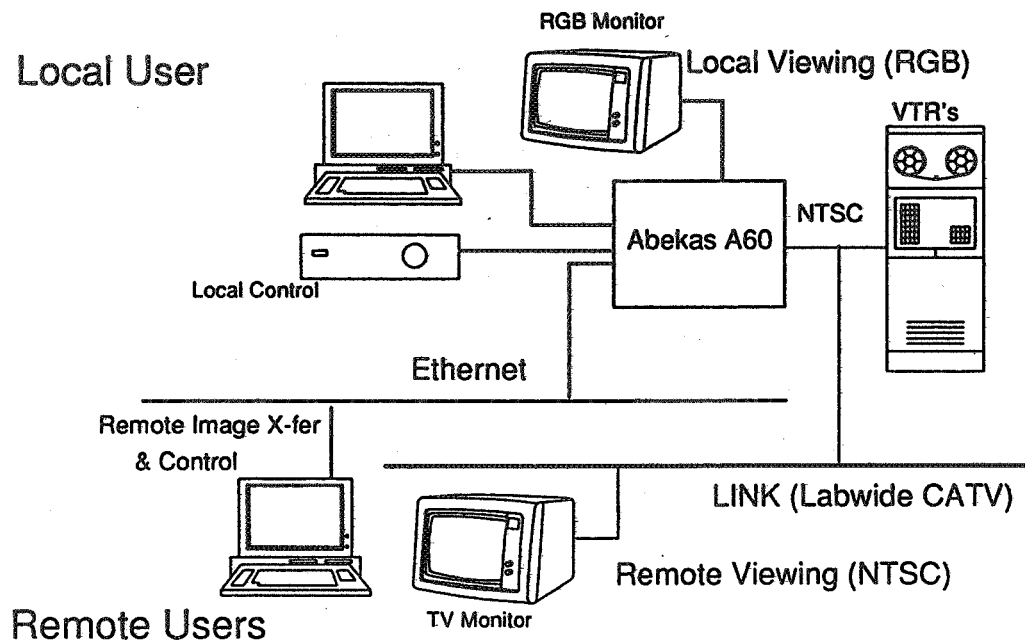
Color Conversion

Translation of the workstation's red-green-blue component color space to the composite NTSC color system can generate various artifacts. This can be a major problem when color is used to convey scientific information. Several techniques are used at LeRC to overcome these problems including: color-table adjustment to avoid over-saturation, and the use of edge-outlined contour plots to emphasize subtle color differences.

Spatial Ambiguities

Since passive viewers are denied the interactivity of workstations, that allows researchers to accurately determine spatial relationships and resolve visual ambiguities, it is critical that recorded data be easily understood. Careful attention must be paid to relative object and 'camera' movements and scene content. In addition, advanced display techniques such as solid, smooth-shaded stream-tubes instead of line-segment tracers can significantly improve flow visualization.

LeRC advanced Graphics & VISualization laboratory (Video Animation Subsystem)



Video as a CFD Research Tool

- * *Visualize Time-Dependent Results*
- * *Record Interactions with Data*
- * *Workstations too slow for complex data*
- * *Video cost effective & ubiquitous*

Successful CFD Video Visualizations:

Technology -

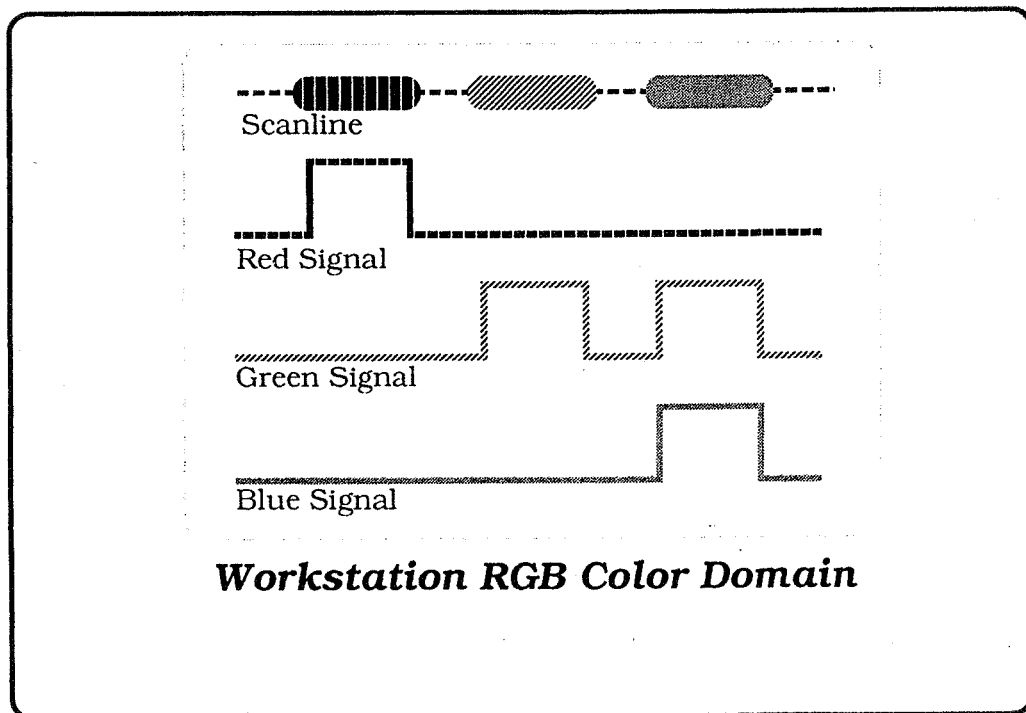
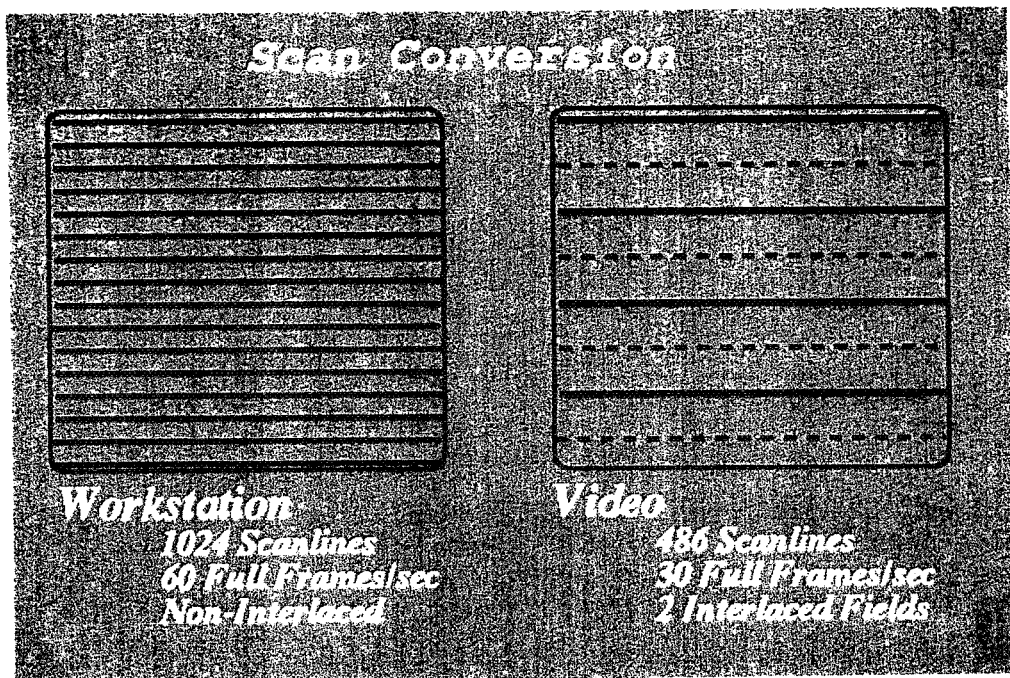
- Scan Conversion*
- Color Conversion*
- Recording media*

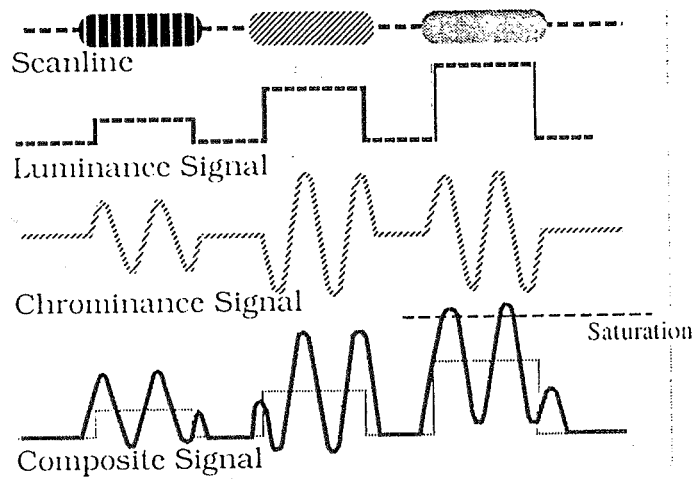
Aesthetics -

- Resolving Spatial Ambiguities*
- Motion Control*
- Script*

Problems:

- * Most researchers are unaware of steps necessary to go from workstation to video*
- * Most visualization software don't provide all necessary tools*
- * 'Video-ifying' the visualization usually occurs at the last minute*

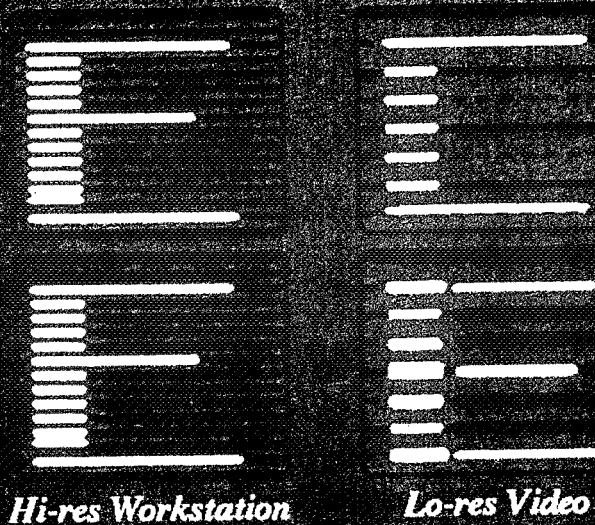




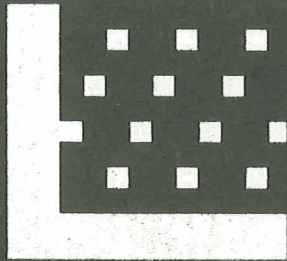
NTSC Color Domain

Susceptible to adjacent pixel color interference

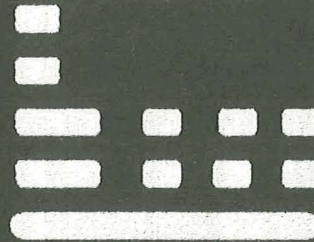
Possible Effects of Scan Conversion



Effect of Scan Conversion on Texture Patterns



Workstation



Video

NTSC Color Artifacts

- * *Zippers*
- * *Dot Crawl*
- * *Saturation*



Image at risk

Modify Colormaps to avoid over-saturation

Luminance:

$$Y = 0.29*r + 0.59*g + 0.11*b$$

Chrominance:

$$I = 0.59*r - 0.27*g - 0.32*b$$

$$Q = 0.21*r - 0.52*g + 0.31*b$$

Saturation:

$$S = (I^2 + Q^2)^{1/2}$$

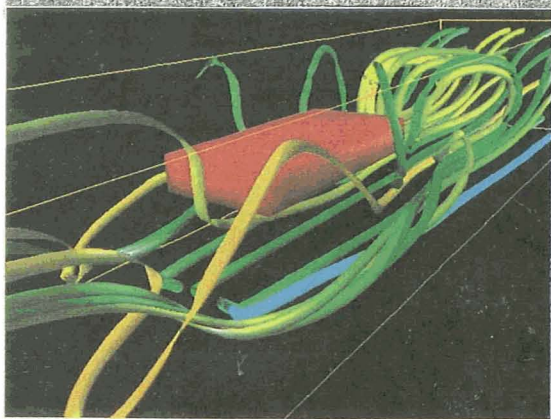
O.K. If $-0.25 < (Y-S)$, $(Y+S) < 1.0$

Contour outlining to avoid NTSC edge artifacts

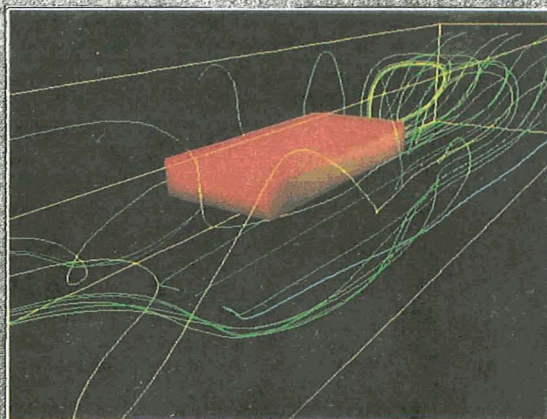


PRECEDING PAGE BLANK NOT FILMED

*Resolve Spatial Ambiguities:
Video is passive -- viewer cannot
manipulate objects.*

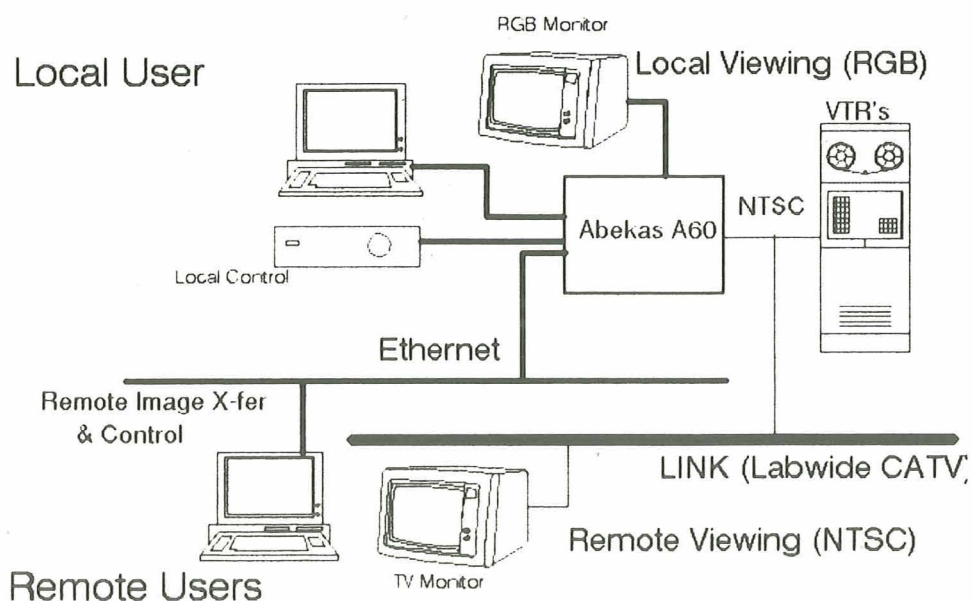


Line tracers



Tube tracers

*advanced Graphics & VISualization laboratory's
digital video facility*



Techniques for CFD Animation

- * Use thick lines when possible***
- * Outline contour edges if possible***
- * Use nonsaturating colors***
- * Use shaded tracers***
- * Preview images on TV monitor***
- * Avoid simultaneous viewpoint and data motion***

PRECEDING PAGE BLANK NOT FILMED

DISTRIBUTED VISUALIZATION FOR COMPUTATIONAL FLUID DYNAMICS

Don J. Sosoka and Anthony A. Facca
Computer Science Division
NASA Lewis Research Center

Distributed concurrent visualization and computation in Computational Fluid Dynamics is not a new concept. Specialized applications such as RIP (Realtime Interactive Particle-tracer) and vendor specific tools like DGL (Distributed Graphics Language) have been in use for some time. Little work however, has been done to put together a concise, easy to use set of routines which would allow the CFD code developer to take advantage of this technology. Recent advances in the speeds of the supercomputer and graphics workstations have made it possible to realize many of the applications for distributed visualization which have long been envisioned. Among these are the ability to "watch" a CFD code as it runs on a supercomputer, to perform interactive grid generation on a local workstation interacting with a solver running on the supercomputer, and ultimately, to "steer" the CFD run interactively from the workstation. For any of these applications to be developed, a set of tools are required to provide this distributed capability to the scientist. Three components can be identified as being necessary to make this capability useful to the CFD researcher. First, a FORTRAN-callable set of network routines must be provided to "link-up" the distributed processes and provide continuous process-to-process communication. Data is moved from the computational process to the visualization process transparently. Secondly, a FORTRAN-callable set of visualization functions must also be provided to the CFD researcher. These functions must allow the researcher to easily view the data in a meaningful, understandable way. Finally, some empirical data should be available to provide guidance to the researcher on the type of applications which would benefit from this kind of distributed processing. Several problems must be addressed in providing each of these three components. In dealing with the network, capacity is always a concern. Binary data compatibility between machines with vastly different architectures is a major factor which must be addressed. Most importantly, ease of use from the CFD code developers point of view must be maintained in delivering a useful solution.

This paper describes a current project underway at NASA Lewis Research Center to provide the CFD researcher with an easy method for incorporating distributed processing concepts into program development. Details on the FORTRAN callable interface to a set of network and the visualization functions are presented along with some results from initial CFD case studies that employ these techniques.

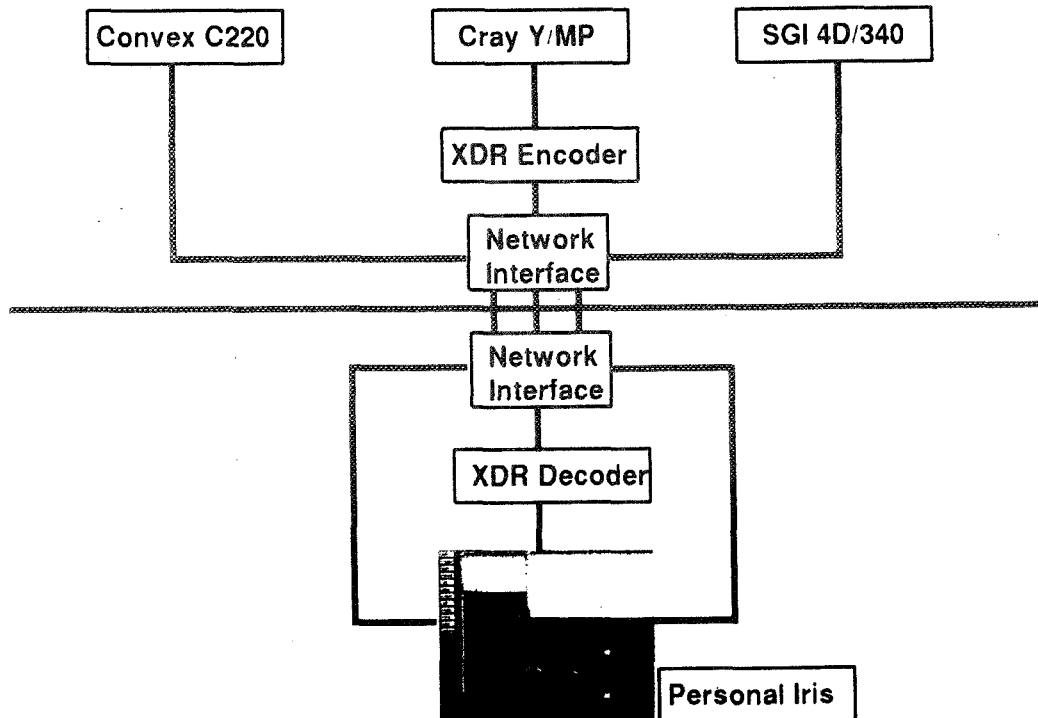
OBJECTIVE:

PROVIDE CFD RESEARCHERS AN EASY METHOD TO
INCORPORATE **DISTRIBUTED VISUALIZATION** TECHNIQUES
INTO THEIR PROGRAM DEVELOPMENT.

DISTRIBUTED VISUALIZATION:

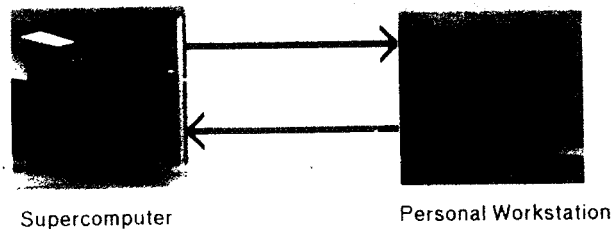
CONCURRENT GRAPHICAL RENDERING ON ONE SYSTEM
OF RESULTS, AS THEY ARE COMPUTED ON ANOTHER.

Distributing Computational Code



APPLICATIONS FOR DISTRIBUTED VISUALIZATION IN CFD

- Convergence / Status Checking (PEEKER)
- Interactive / Batch Grid Generation
- Immediate Visualization of Time-Dependent Calculations
- Interactive "Steering" of Remote Computations



DEGREES OF DISTRIBUTED VISUALIZATION

- All computation, scientific visualization, and graphics performed remotely.
ex: Ultra Frame Buffer
- Computations and scientific visualization done remotely, graphics performed by the workstation.
ex: DGL for SGI workstations
- Numeric Computations done remotely, all scientific visualization and graphics performed by the workstation.
most difficult to implement
 - requires network interface capable of transferring transparent binary datahighest payoff
 - provides greatest degree of parallelism
 - no static datasets required to visualize results

TOOLS REQUIRED FOR DISTRIBUTED VISUALIZATION

- **FORTTRAN-Callable Set of Network Interface Routines**
 - establish network connection between processes
 - provide continuous process-to-process communication

- **FORTTRAN-Callable Set of Visualization Functions**
 - incorporate state-of-the-art rendering technology
 - utilize non-proprietary graphics interface
 - encompass all current CFD post-processing capability

**Design Goal: Make Routines Easy to Use From a CFD Code
Developers Point of View**

Open Close Routines:

OPNCON -- Open a "connect" on a port
OPNLIS -- Open a "listen" on a port

Send Routines:

PUTRA1 -- Send array of real numbers
PUTIA1 -- Send array of integer numbers
PUTSTR -- Send character string
PUTRA2 -- Send 2-dimensional array real numbers
PUTIA2 -- Send 2-dimensional array of integers

Receive Routines:

GETRA1 -- Receive array of real numbers
GETIA1 -- Receive array of integer numbers
GETSTR -- Receive character string
GETRA2 -- Receive 2-dimensional array real numbers
GETIA2 -- Receive 2-dimensional array of integers

```

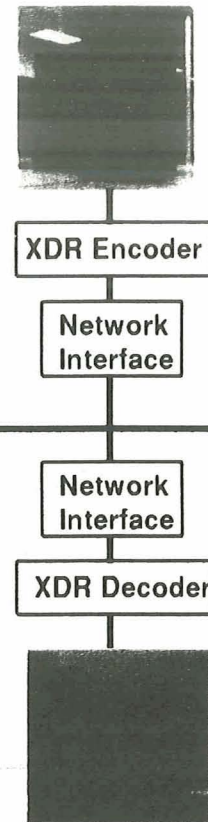
c Sample SEND program..
:
c establish network connection
CALL OPNCON( ... )
:
c transfer 2D computational array
CALL PUTRA2( ... )
:

```

```

c Sample RECEIVE program..
:
c accept network connection
CALL OPNLIS( ... )
:
c receive 2D array from server
CALL GETRA2( ... )
:

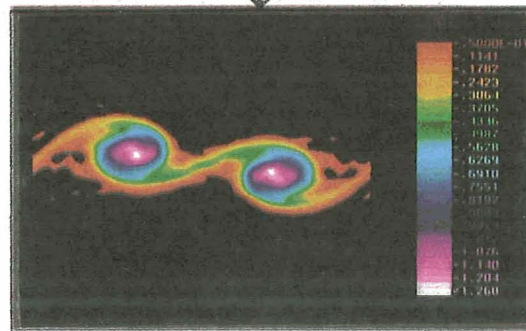
```



```

c *** SVP/NETWORK RECEIVE CODE
PARAMETER (ID=80,JD=70)
REAL      xgrd(ID,JD),ygrd(ID,JD)
REAL      ws(ID,JD)
c *** access the network
call OPNLIS
c *** get data from the network
100 CONTINUE
    call GETIA1(1,nx,istat)
    call GETIA1(1,ny,istat)
    call GETRA2(nx,ny,2,ws,istat)
    call GETRA2(nx,ny,2,xgrid,istat)
    call GETRA2(nx,ny,2,ygrid,istat)
c *** introduce data to SVP
    call GDATA(nx,ny,nx,ny,xgrid,ygrid)
    call FDATA(ws,' ')
    call GENCON(' ')
    call CLRSVP
GOTO 100
STOP
END

```



TECHNIQUES FOR GRID MANIPULATION AND ADAPTATION

Yung K. Choo
Internal Fluid Mechanics Division
NASA Lewis Research Center

Peter R. Eisemann
Program Development Corporation
and
Ki D. Lee
University of Illinois

Two approaches have been taken to provide systematic grid manipulation for improved grid quality.

One is the control point form (CPF) of algebraic grid generation. It provides explicit control of the physical grid shape and grid spacing through the movement of the control points. The control point array, called a control net, is a sparse grid-type framework in physical space. The CPF-based grid manipulation has many merits. The method is efficient and easy to implement, since its formulation is algebraic and concise. Grid quality can be easily enhanced by numerous local and global grid distribution strategies. It is also compatible with various complementary operations needed in a quality grid-generation procedure. It works well in the interactive computer graphics environment and hence can be a good candidate for integration with other emerging technologies.

The other approach is grid adaptation using a numerical mapping between the physical space and a parametric space. Grid adaptation is achieved by modifying the mapping functions through the effects of grid control sources. The source strengths are extracted from the distribution of flow or geometric properties on the initial grid. Grids can be made adaptive to geometry, flow solution, or grid quality, depending on what property is used to define the source strengths. One advantage of the method is that the basic characteristics of the initial grid can be retained while adapting it to grid quality. The use of grid control sources allows for linear combinations of different controls based on the superposition principle. And the grid can be adapted to more than one property through a series of mappings. The source formulation promotes smooth variations in the grid, even with irregularly distributed sources. The adaptation process can be repeated in a cyclic manner if satisfactory results are not achieved after a single application.

PRECEDING PAGE BLANK NOT FILMED

OUTLINE

1. EXPLICIT GRID MANIPULATION BY USER

TECHNICAL APPROACH - CONTROL POINT FORM (CPF)

OBJECTIVES / ADVANTAGES

STATUS / RESULTS

FUTURE DIRECTION

2. GRID ADAPTATION TO SOLUTIONS, QUALITY

TECHNICAL APPROACH - NUMERICAL MAPPING
with GRID CONTROL SOURCE

OBJECTIVES / ADVANTAGES

STATUS / RESULTS

FUTURE DIRECTION

OBJECTIVES

TO DEVELOP EASY, EXPLICIT GRID MANIPULATION CAPABILITY

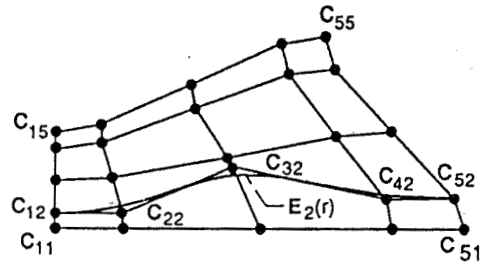
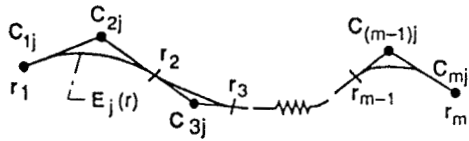
ADVANTAGES

THE CPF-BASED GRID GENERATION:

- IS EFFICIENT AND EASY TO IMPLEMENT
- CAN EASILY ENHANCE GRID QUALITY BY VARIOUS LOCAL/GLOBAL GRID DISTRIBUTION STRATEGIES
- CAN BE USED AS A COMPLEMENTARY TOOL AS WELL AS A STAND-ALONE TOOL
- WORKS WELL IN THE INTERACTIVE COMPUTER GRAPHICS ENVIRONMENT
- ALLOWS TO MANIPULATE GEOMETRY (FREE-FORM BOUNDARY)
- MAKES INTERACTIVE CFD ATTRACTIVE

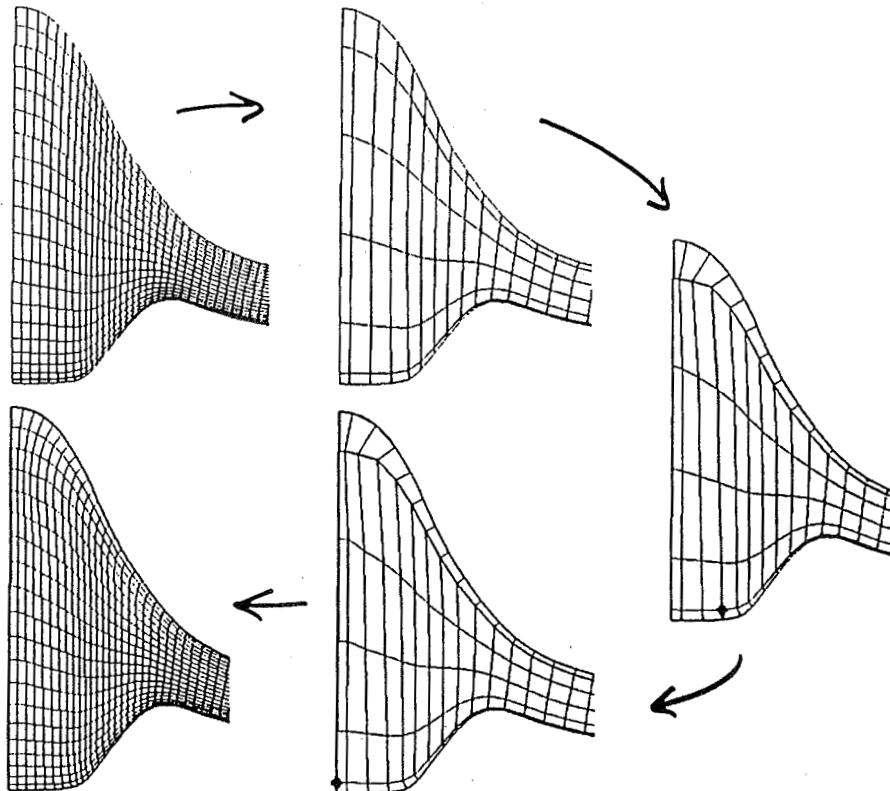
TECHNICAL APPROACH - CONTROL POINT FORM (CPF)

CONSTRUCTION OF A CURVE & A CONTROL NET

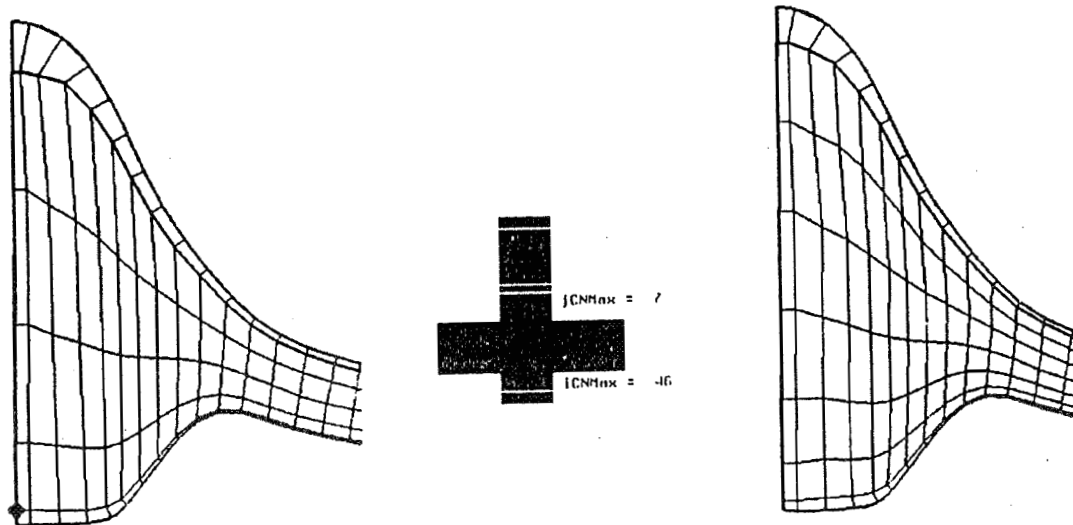


$$\begin{aligned}
 Q(r, t) = & T(r, t) + \alpha_1 [1 - G_1(r)] [P(1, t) - F_1(t)] \\
 & + \alpha_2 G_{N-1}(r) [P(N-1, t) - F_N(t)] \\
 & + \alpha_3 [1 - H_1(t)] [P(r, 1) - E_1(r)] \\
 & + \alpha_4 H_{M-1}(t) [P(r, M-1) - E_M(r)]
 \end{aligned}$$

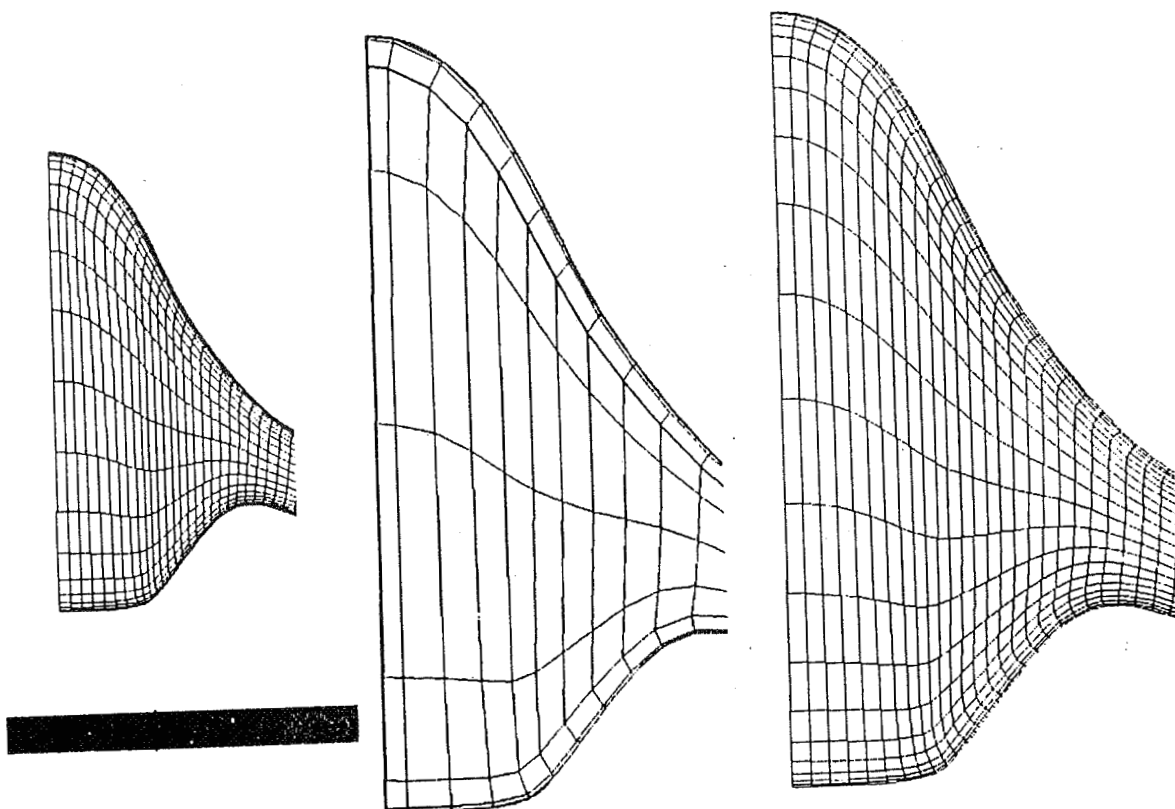
TURBO/I INTERACTIVE PROCESS



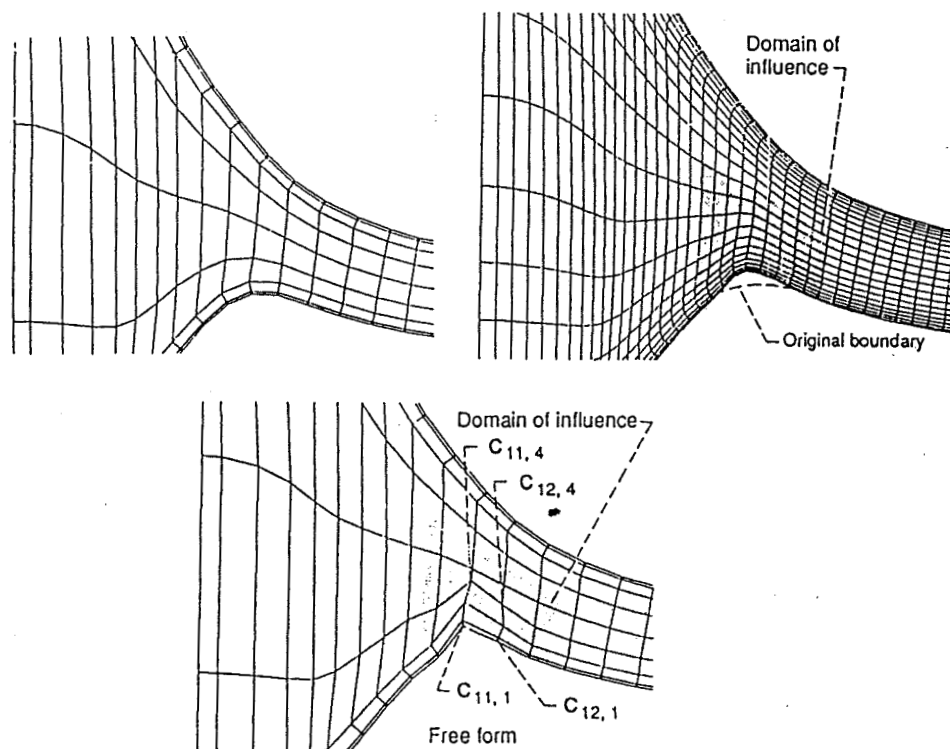
TURBO/I INTERACTIVE PROCESS - CONTINUED
CHANGING THE NUMBER OF CONTROL POINTS



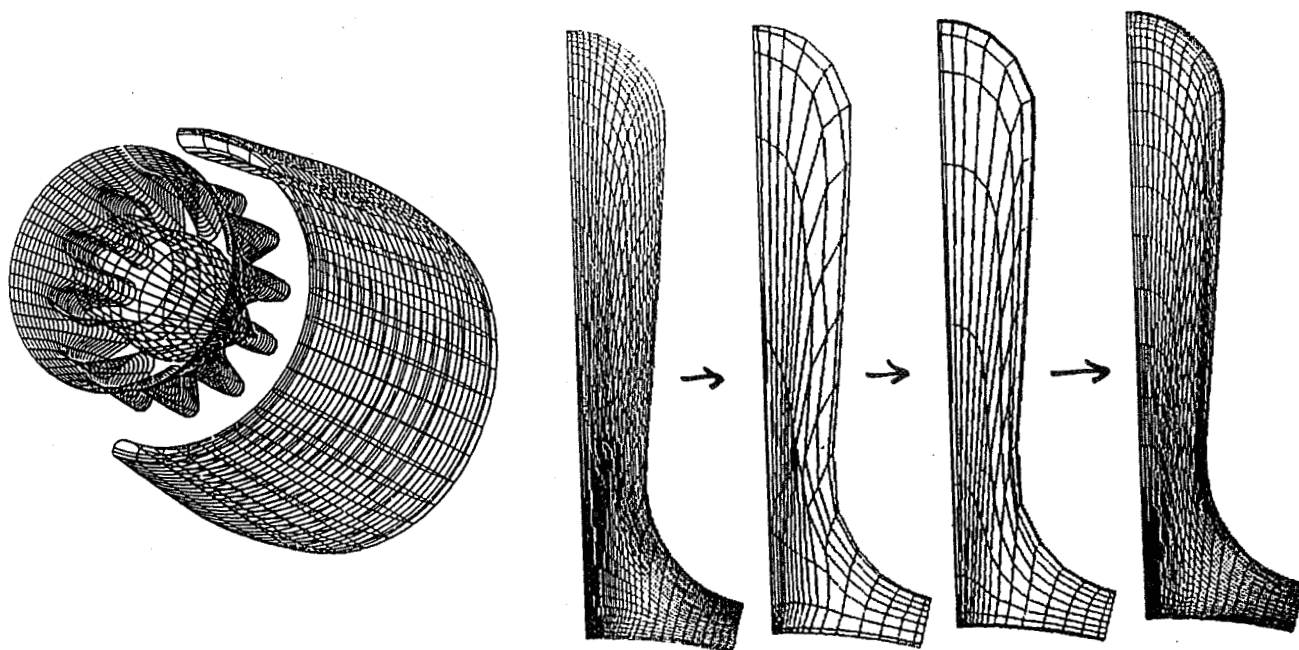
TURBO/I INTERACTIVE PROCESS - CONTINUED
STRETCHING



GEOMETRY MODIFICATION USING FREE-FORM BOUNDARY



TURBO/I AS A COMPLEMENTARY TOOL



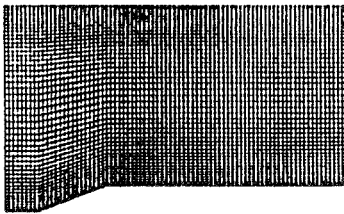
FUTURE DIRECTION

SURFACE GRID CONTROL USING CONTROL POINT FORM

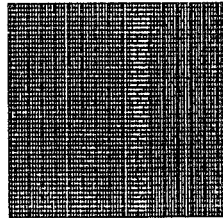
INTERACTIVE ADAPTIVE GRID GENERATION

TECHNICAL APPROACH

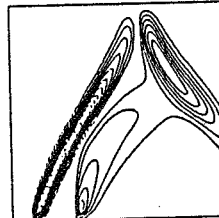
NUMERICAL MAPPING WITH GRID CONTROL SOURCE



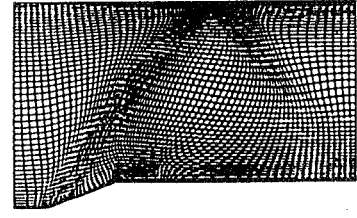
initial grid



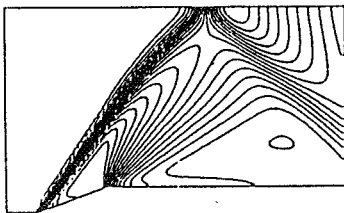
(s,t) parametric domain



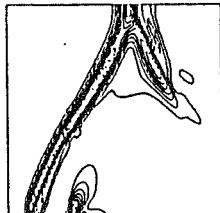
σ' distribution



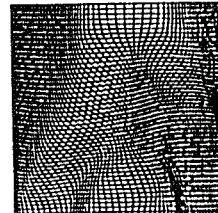
solution-adapted grid



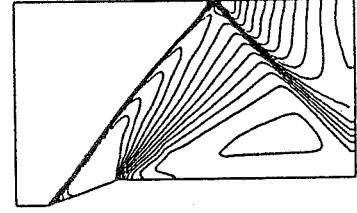
isobars on initial grid



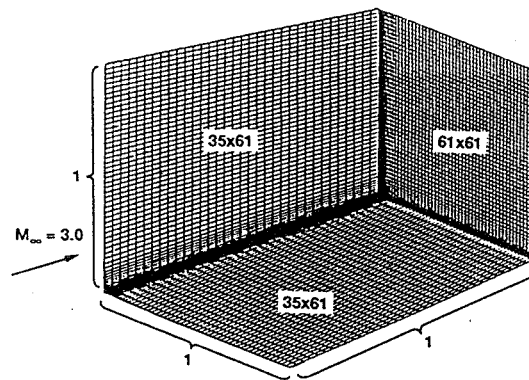
σ' distribution



(s',t') parametric domain



isobars on adapted grid



Grid control sources definition:

$$\sigma_{kl}^s = w_0^s |\phi| + w_1^s \left| \frac{\partial \phi}{\partial s} \right| + w_2^s \left| \frac{\partial^2 \phi}{\partial s^2} \right|$$

$$\sigma_{kl}^t = w_0^t |\phi| + w_1^t \left| \frac{\partial \phi}{\partial t} \right| + w_2^t \left| \frac{\partial^2 \phi}{\partial t^2} \right|$$

where w_i = weighting factors

ϕ = a grid quality parameter

Mapping modification:

$$s'_{ij} = s_{ij} + \sum_{k,l} K_{ijkl}^s \sigma_{kl}^s$$

$$t'_{ij} = t_{ij} + \sum_{k,l} K_{ijkl}^t \sigma_{kl}^t$$

where (s', t') = modified parametric coordinates

K_{ijkl}^s, K_{ijkl}^t = influence coefficients

OBJECTIVES

TO DEVELOP GRID ADAPTATION CAPABILITY TO SOLUTIONS, GRID QUALITY,
AND GEOMETRY TO OBTAIN ACCURATE PREDICTION OF COMPLEX FLOWS

ADVANTAGES

THE APPROACH IS EASY TO IMPLEMENT AS ONE OF THE REDISTRIBUTION
SCHEME.

THE USE OF GRID CONTROL SOURCES ALLOWS TO ADAPT THE GRID BY USING
LINEAR COMBINATIONS OF FLOW PROPERTIES.

THE APPROACH CONTROLS BOTH GRID DENSITY AND QUALITY WITHOUT
ANY ADVERSE INFLUENCE OF ONE TO THE OTHER.

THE GRID ADAPTATION SCHEME CAN BE INTEGRATED WITH WELL-DEVELOPED
FLOW SOLVERS AS SUBROUTINES.

NEEDS

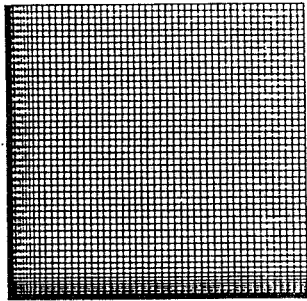
FULL 3D ADAPTATION WITH REDUCED COST

STATUS / RESULTS

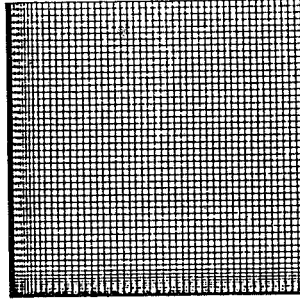
2D ADAPTATION TO SOLUTIONS, GRID QUALITY

APPLICATION OF 2D ADAPTATION TECHNIQUE TO 3D PROBLEMS

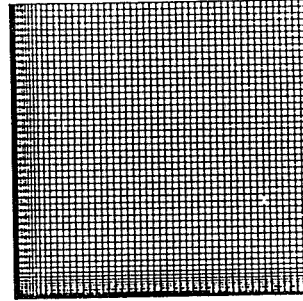
CORNER FLOW GRID COMPARISON



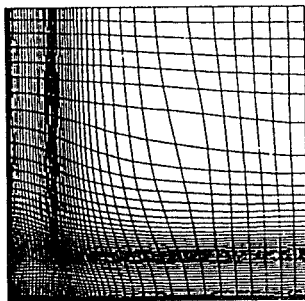
Initial grid
@ $x=0.407$



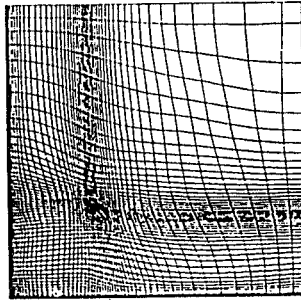
Initial grid
@ $x=0.688$



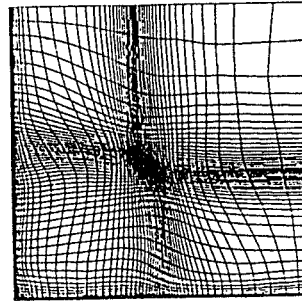
Initial grid
@ $x=1.000$



Adapted grid
@ $x=0.407$

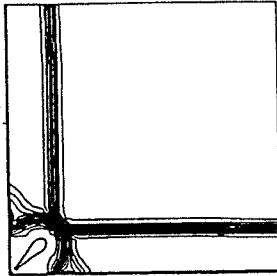


Adapted grid
@ $x=0.688$

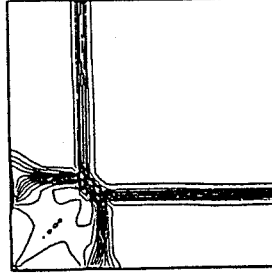


Adapted grid
@ $x=1.000$

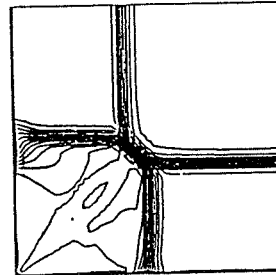
CORNER FLOW SOLUTION COMPARISON



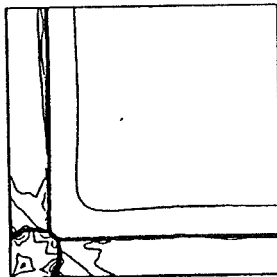
Density contours
on initial grid
@ $x=0.407$



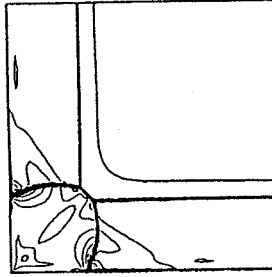
Density contours
on initial grid
@ $x=0.688$



Density contours
on initial grid
@ $x=1.000$



Density contours
on adapted grid
@ $x=0.407$



Density contours
on adapted grid
@ $x=0.688$



Density contours
on adapted grid
@ $x=1.000$

FUTURE DIRECTION

EXTEND IT TO FULL 3D ADAPTATION

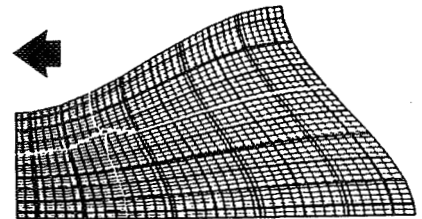
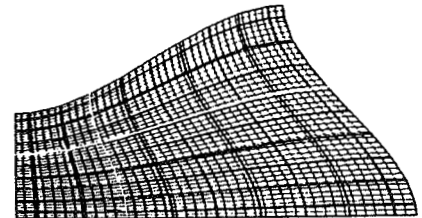
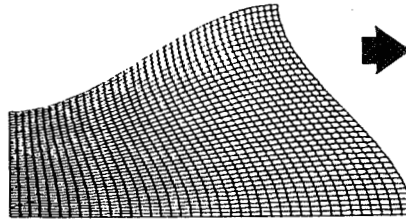
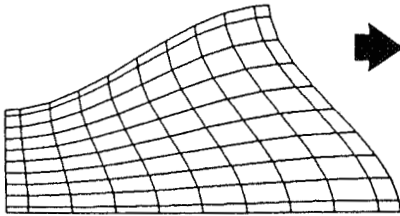
EXPLORE AND IMPLEMENT COST REDUCTION SCHEMES

EXPLORE POSSIBLE BENEFITS FROM PARALLEL COMPUTING

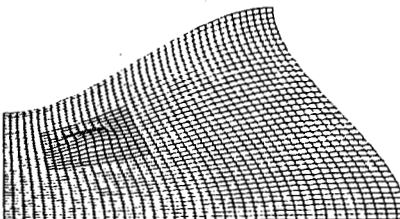
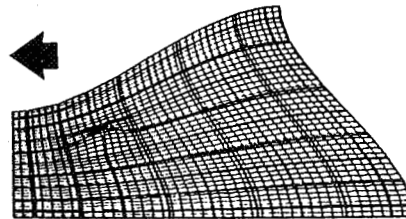
APPLY IT TO CHALLENGING PROPULSION PROBLEMS

INTERACTIVE GRID GENERATION

Program TURBO



TRANSLATE THE CONTROL POINT



EXAMINE NEW GRID
-ACCEPT OR REPEAT THE PROCESS

COMPUTE NEW GRID

COMPUTE & EXAMINE INITIAL
GRID

SELECT A CONTROL POINT TO
MODIFY GRID

CONSTRUCT A SIMPLE CONTROL NET
GENERATE SURFACE GRID

REPORT DOCUMENTATION PAGEForm Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE February 1992	3. REPORT TYPE AND DATES COVERED Conference Publication	
4. TITLE AND SUBTITLE Computational Fluid Dynamics			5. FUNDING NUMBERS WU-505-62-52	
6. AUTHOR(S)				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Lewis Research Center Cleveland, Ohio 44135-3191			8. PERFORMING ORGANIZATION REPORT NUMBER E-6374	
9. SPONSORING/MONITORING AGENCY NAMES(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, D.C. 20546-0001			10. SPONSORING/MONITORING AGENCY REPORT NUMBER CP-10078	
11. SUPPLEMENTARY NOTES Responsible person, Dr. Louis Povinelli, (216) 433-5818.				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified - Unlimited Subject Categories 07 and 20			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) This collection of papers was presented at the Computational Fluid Dynamics (CFD) Conference held at Ames Research Center in California on March 12 through 14, 1991. It is an overview of CFD activities at NASA Lewis Research Center. The main thrust of computational work at Lewis is aimed at propulsion systems such as gas turbines, rocket engines, and hypersonic propulsion systems. Specific issues related to propulsion CFD and associated modeling will also be presented. Examples of results obtained with the most recent algorithm development will also be presented.				
14. SUBJECT TERMS CFD, Numerical methods; Propulsion CFD, Algorithm development			15. NUMBER OF PAGES 242	
			16. PRICE CODE A11	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT	

National Aeronautics and
Space Administration

Lewis Research Center
Cleveland, Ohio 44135

Official Business
Penalty for Private Use \$300

FOURTH CLASS MAIL

ADDRESS CORRECTION REQUESTED



Postage and Fees Paid
National Aeronautics and
Space Administration
NASA 451

NASA
